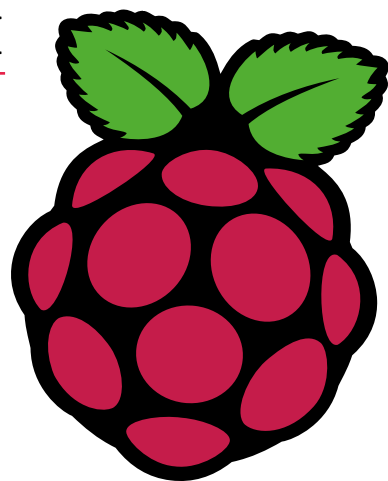




BUY IN PRINT **WORLDWIDE** MAGPI.CC/STORE

The MagPi



Issue 97 | September 2020 | magpi.cc

The official Raspberry Pi magazine



Play PC
classics

3D Printer



3D



The best
starter bundle



Raspberry Pi

Printing

&

Making

with Raspberry Pi



Hear
world music
with RadioGlobe

Design, craft and build
your own accessories



Laser Cutter



GLOBAL
DELIVERY
magpi.cc/store

46 PAGES OF PROJECTS & TUTORIALS

9.2 MILLION+ PRODUCTS ONLINE | 1,100+ INDUSTRY-LEADING SUPPLIERS | 1.9 MILLION+ PRODUCTS IN STOCK



YOU INNOVATE. WE CAN HELP. **DIGIKEY.CO.UK**



The World's Largest Selection of Electronic Components Available for Immediate Dispatch™

*A shipping charge of £12.00 will be billed on all orders of less than £33.00. A shipping charge of \$18.00 USD will be billed on all orders of less than \$50.00 USD. All orders are shipped via UPS, Federal Express, or DHL for delivery within 1-3 days (dependent on final destination). No handling fees. All prices are in British pound sterling or United States dollar. Digi-Key is a franchised distributor for all supplier partners. New products added daily. Digi-Key and Digi-Key Electronics are registered trademarks of Digi-Key Electronics in the U.S. and other countries. © 2020 Digi-Key Electronics, 701 Brooks Ave. South, Thief River Falls, MN 56701, USA

ECIA MEMBER
Supporting The Authorized Channel

WELCOME

to *The MagPi* 97

If you want to love something, make it yourself. It's not a bad maxim. Making your own computers, accessories, robots, and widgets is a hugely rewarding experience.

This month's feature, 3D Printing & Making with Raspberry Pi (page 32) is a team effort by two makers I admire: PJ Evans, a regular contributor to these pages, and Tim Richardson from Pi Wars. Together, PJ and Tim will walk you through the materials, equipment, and processes you need to make your kit and how to get the most out of making with Raspberry Pi.

Elsewhere, in this magazine there are great examples of *The MagPi* readers making incredible projects. Take a look at RadioGlobe (page 8), a globe that's been hacked to play radio from around the world; or the Oktoberfest Pinball Machine you can play for real, but online (page 18). Two of my favourite pieces of kit, AIY Projects and Coral, are also back (page 28).

These are projects you can't find anywhere else; you have to build them.

Learn to make and a whole world will open up! So let's make it happen.

Lucy Hattersley Editor



EDITOR

Lucy Hattersley

Lucy is editor of *The MagPi* and likes making robots. This month she enjoyed the CrowPi2 kit (page 76).

@LucyHattersley





The **MagPi**

HackSpace
TECHNOLOGY IN YOUR HANDS

CustomPC

3 ISSUES FOR £10



FREE BOOK



magpi.cc/freebook

Subscribe to The MagPi, HackSpace magazine, or Custom PC. Your first three issues for £10, then our great value rolling subscription afterwards. Includes a free voucher for one of five fantastic books at store.rpiexpress.co.uk/collections/latest-bookazines
UK only. Free delivery on everything.

Contents

► Issue 97 ► September 2020

Cover Feature

32 3D Printing & Making

Regulars

- 92 Your Letters
- 97 Next Month
- 98 The Final Word

Project Showcases

- 08 RadioGlobe
- 14 Retro gaming NES frame
- 16 Atomic TV
- 18 Oktoberfest Pinball Machine
- 22 Retro TV Arcade Machine
- 26 Black Lives Matter projection



Atomic TV



Oktoberfest Pinball Machine

DISCLAIMER: Some of the tools and techniques shown in The MagPi magazine are dangerous unless used with skill, experience, and appropriate personal protection equipment. While we attempt to guide the reader, ultimately you are responsible for your own safety and understanding the limits of yourself and your equipment. Children should be supervised. Raspberry Pi (Trading) Ltd does not accept responsibility for any injuries, damage to equipment, or costs incurred from projects, tutorials or suggestions in The MagPi magazine. Laws and regulations covering many of the topics in The MagPi magazine are different between countries, and are always subject to change. You are responsible for understanding the requirements in your jurisdiction and ensuring that you comply with them. Some manufacturers place limits on the use of their hardware which some projects or suggestions in The MagPi magazine may go beyond. It is your responsibility to understand the manufacturer's limits.

Tutorials

- 40** DOSBox-X legal game emulation
- 44** HQ Camera: flash photography
- 48** Home music system – part 2
- 52** Home monitoring dashboard
- 56** Connect Raspberry Pi 4 to iPad
- 62** Tap-an-LED game – part 1
- 66** Code Gauntlet's four-player co-op

The Big Feature



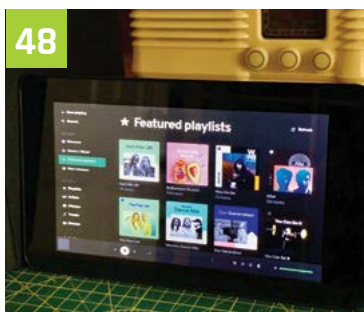
Make games on Raspberry Pi

Reviews

- 76** CrowPi2
- 78** M.A.R.S. Rover
- 80** 10 best starter kits
- 82** Learn Scratch 3

Community

- 84** Alan O'Donohoe interview
- 86** This month in Raspberry Pi
- 90** Events calendar



Home music system part 2



Connect Raspberry Pi 4 to iPad



CrowPi2



Alan O'Donohoe interview

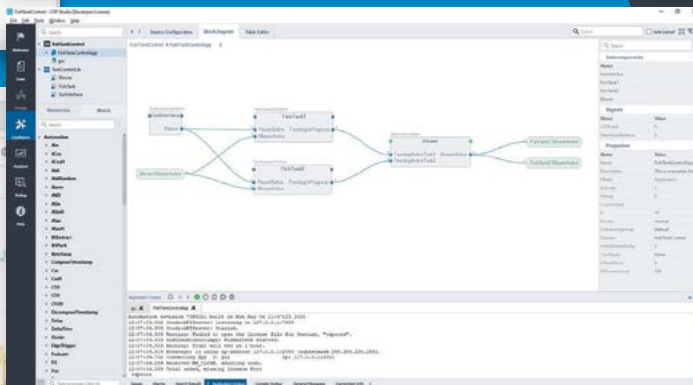
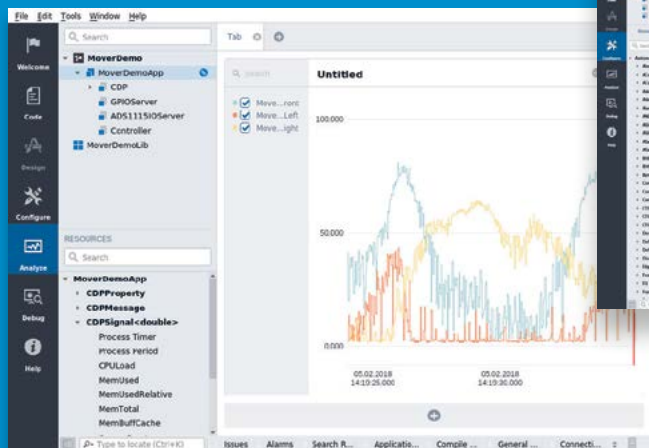
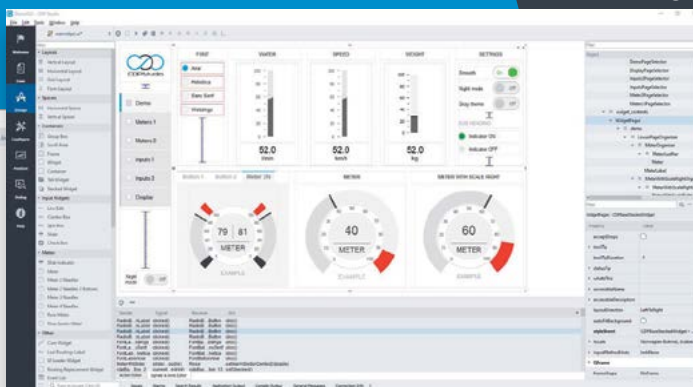
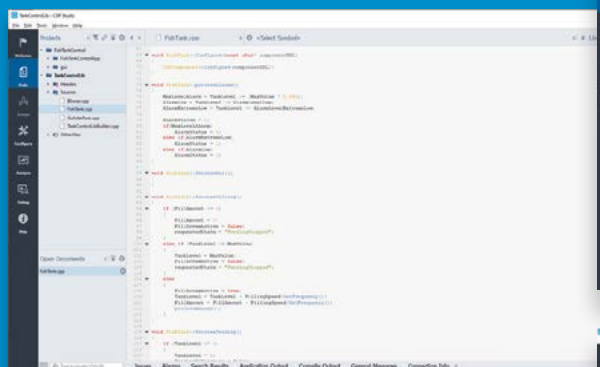
WIN
1 OF 5

PICAR-V V2.0 KITS!

95

Design

Code



Configure

Analyze

CDP STUDIO: HOME PROJECTS MADE EASY

Professional control system development tool

CDP Studio is a great software development tool for your home projects. Build systems for Raspberry Pi, use C++, open source libraries, out of the box support for GPIO, I2C, MQTT, OPC UA and more. Create user interfaces in the integrated designer. Code less and do more!

NOW FREE FOR NON-COMMERCIAL USE.

CDP Technologies AS
Hundsværgata 8, 6008 Ålesund, Norway
Tel: +47 990 80 900 • info@cdptech.com
www.cdpstudio.com



RadioGlobe

Want to know what's playing on the radio anywhere in the world? **Rob Zwetsloot** goes on tour with RadioGlobe



Jude Pullen

Self-proclaimed 'prototyping expert' and 'technologist' with an impressive resume of work designing concepts for Dyson, Sugru, Lego, DesignSpark, and even TV.

magpi.cc/radioglobe



Alert! Drill safety

Please use safety goggles and a desk clamp when drilling.

magpi.cc/drillsafety

People sometimes ask us 'why?' when we share projects online. Why make something that already exists in an easier format, or isn't very functional? The simple answer is: because we can. It can be very fun to make stuff, and the RadioGlobe is an excellent example of this.

"It's a globe which you spin to search web radio stations," creator Jude Pullen summarises. "It's really simple to use (even my four-year-old son gets it!), but the tech inside is pretty clever to make it work. It has a very special component called a rotary encoder, which has the ability to know the position of a rotating axis (in this case the longitude and latitude) to an accuracy of 0.3 degrees. It means we can navigate to all the major cities in the world and listen to their local stations."

Bringing radio to life

The idea for the project arose through a combination of concepts that were on Jude's mind at once. "I think I'd been reading books about space travel, and the emergence of communications while working on a Channel 4 show called *David Jason's Great British Inventions*," he says. "My job was to build a replica of the Bell telephone (using some scrap wood, piping, wires, tracing paper, and vinegar!), and it kinda blew my mind to consider the advent of this technology in the world. This led me to explore a lot of technology we take for granted, like radio. I was discussing this with a friend who then mentioned a website called RadioGarden [radio.garden] which allows you to look up all radio stations on Google Earth. It's a clever mash-up of two powerful technologies, and I guess my 'build' on this was to make it physical."

Information about where you're pointing and what radio station you're listening to are listed on the display





The rotary encoders are set so they know the position of the reticule over the globe

Quick FACTS

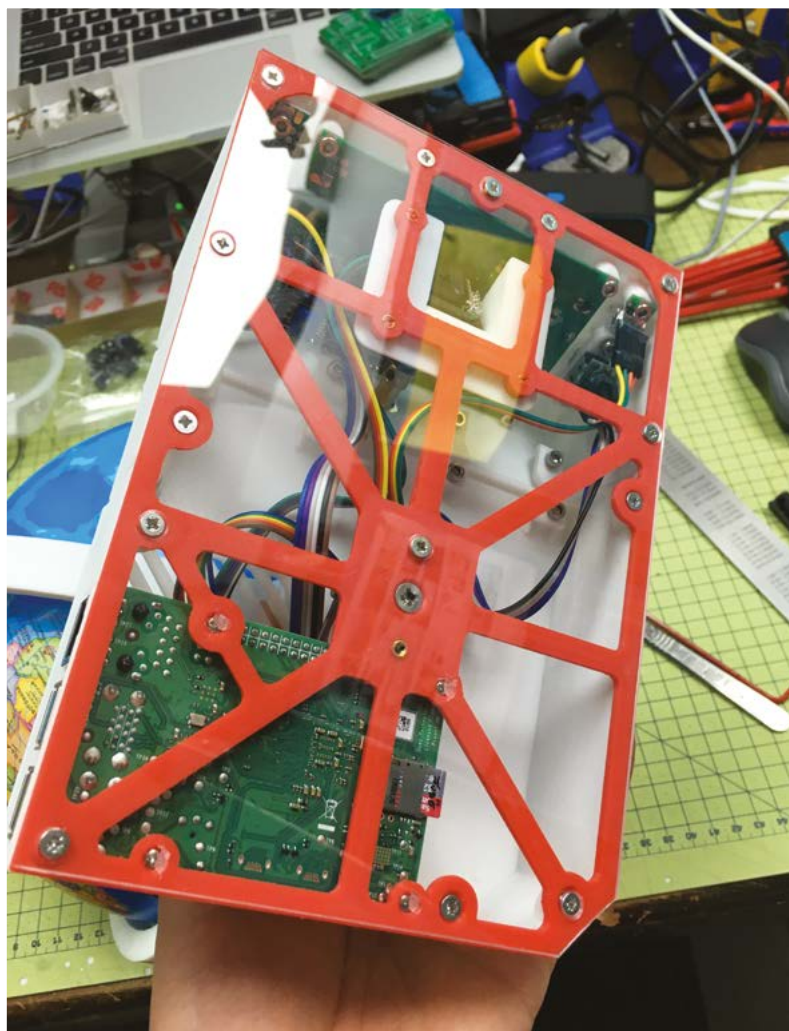
- ▶ Jude usually works with Arduino, but Raspberry Pi was more suited to this project
- ▶ The basic design was created in cardboard first...
- ▶ ... before being 3D-printed in Jude's home workshop
- ▶ Jude's friend Don Robson helped with the code
- ▶ The rotary encoders remember where they are when you next turn it on!

The globe spins just like a toy version!

16.86N, 99.88W
Acapulco, MX
< XHKJ Radio Vida >

RADIOGLOBE DESIGNSPARK RS

- ▶ Just a few of the tools and items you need to build such a device
- ▼ A Raspberry Pi and all the other main electronics are stored in the base



Proof is in the concept

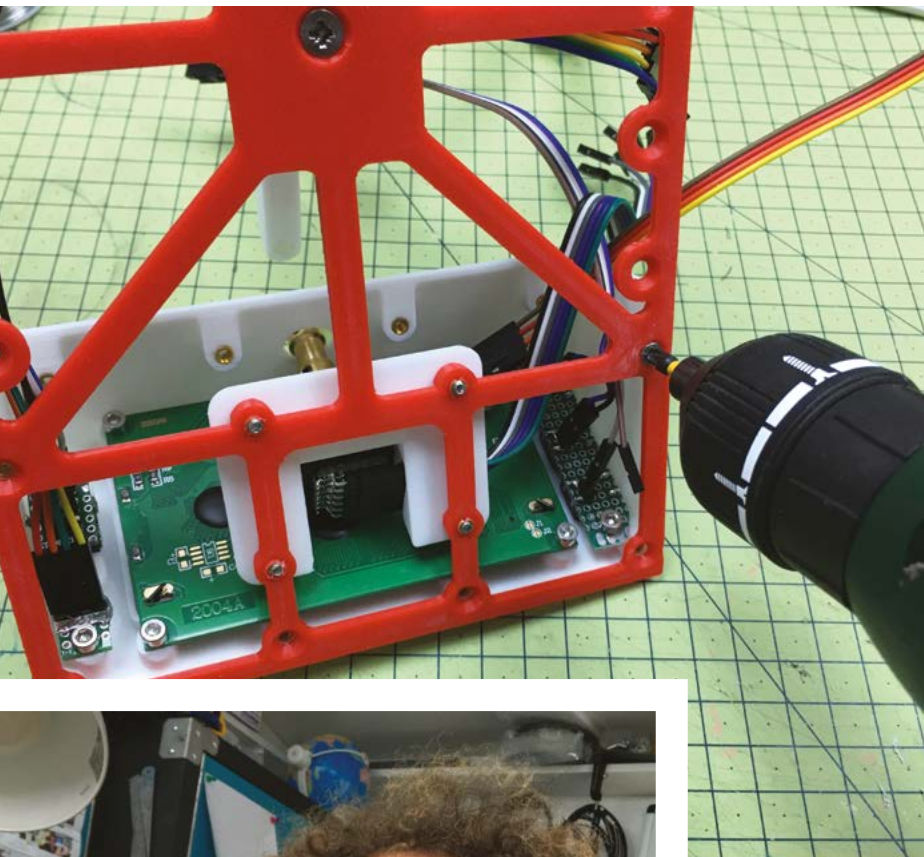
The world is a big place – Jude tells us there are 44,000 radio stations around the world, and not all of them have great broadcast quality. For what he's managed to achieve, he seems pretty happy, though.

"There is a phrase in design called 'proof of concept' or in code it's sometimes called 'minimum viable product,'" he notes. "RadioGlobe is that in terms of its physical design (3D-printed)

" We can navigate to all the major cities in the world and listen to their local stations **"**

and code (V1.0 on GitHub). It works really well in that most of the 2000-odd radio stations do play, but yes there are bugs – in that some radio station links are in countries where broadcast is not perfect, so this can 'trip up' the code, and it stops playing."

Jude has some updates in the works for the RadioGlobe, some of which sound fairly obvious and genuinely useful, such as a recall function for favourites in case you stumble upon a K-pop station you really like, as well as a Shazam button to identify the latest Austrian hit.

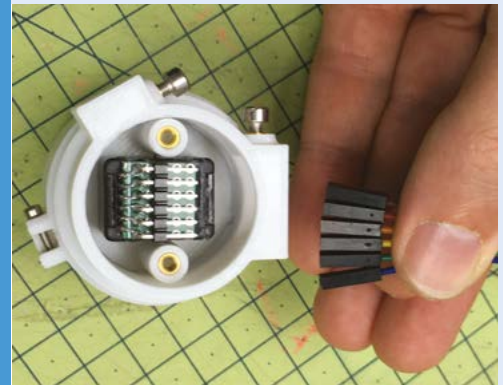


- ▲ The electronics are hidden from the top but visible from below
- ◀ Final adjustments

Find a radio station



- 01** Spin the globe – this is your longitude control – to get to a rough area. Then use the arm to centre the reticle on an exact location. A light will turn from flashing white to red when it's got a location.

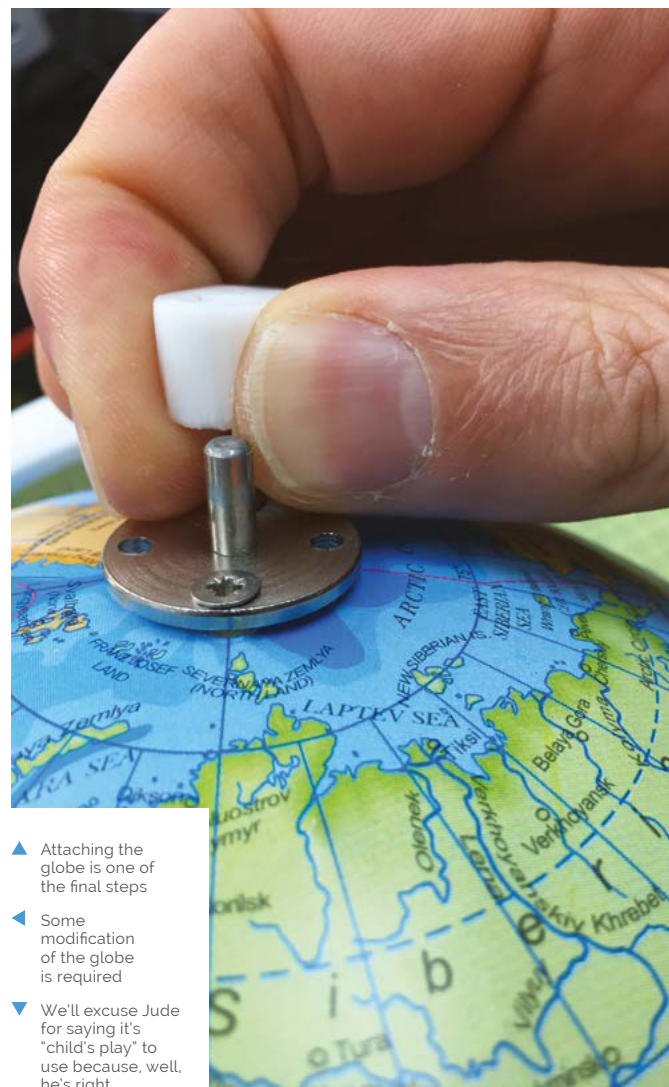


- 02** Two rotary encoders are used to determine the position of the reticle on the globe (that longitude and latitude again), which checks against locations. This turns the light to red when it syncs up with somewhere with playable radio stations.



- 03** You can scroll through a list of stations at the location, but otherwise it will display what the station is on the screen and start playing from it.

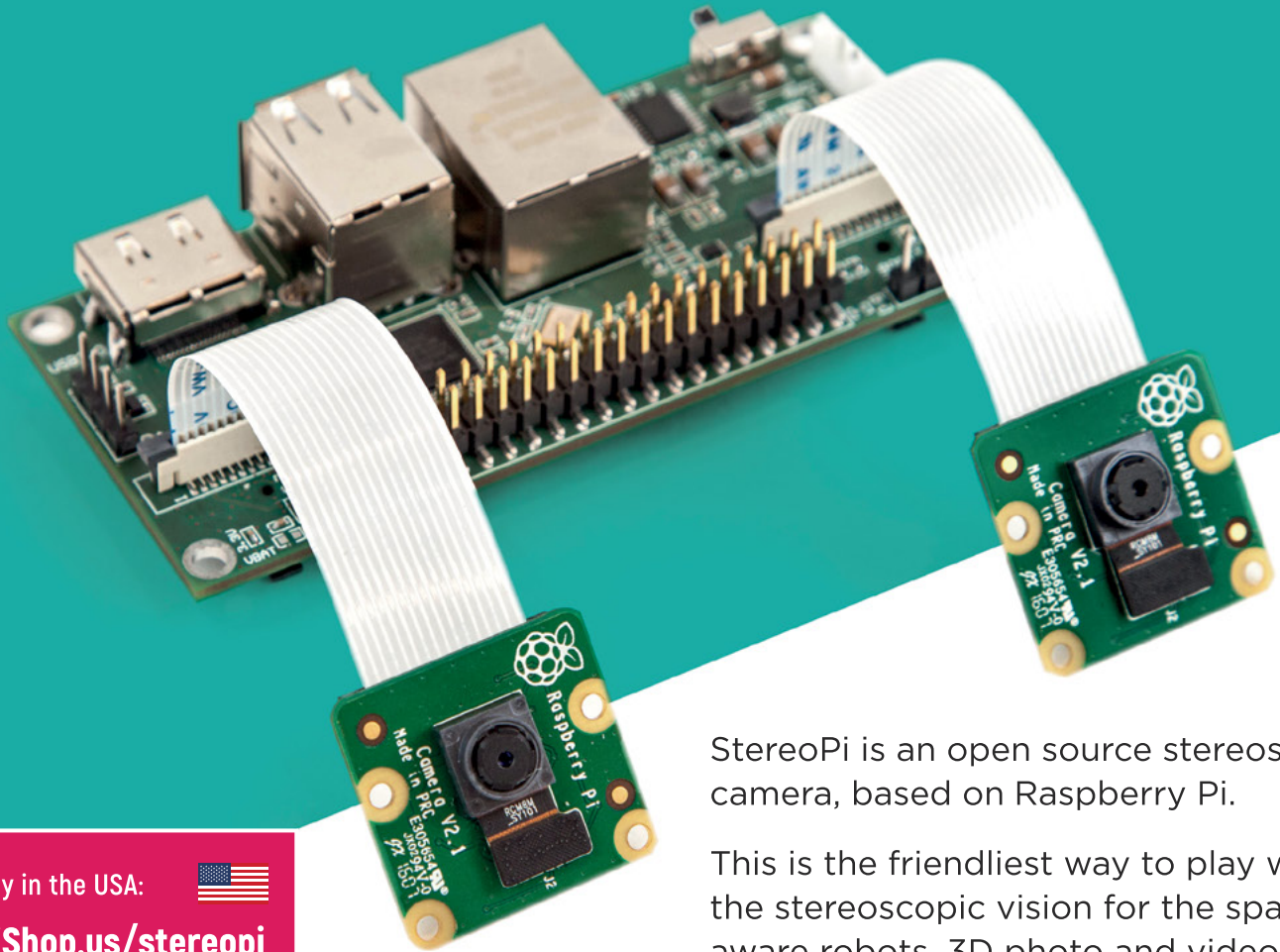
“This is possible because the project is open source, and the code can be contributed to be total strangers from around the world,” Jude explains. “I love the poetic-loop of this! And this has very much been the best of my experience in working with exciting companies like DesignSpark and Raspberry Pi who have championed new ideas that literally might come from anyone, anywhere in the world. Perhaps we’ll have RadioUniverse one day! I joke, but of course RadioGlobe would have seemed crazy 100 years ago too.” [M](#)



- ▲ Attaching the globe is one of the final steps
- ▲ Some modification of the globe is required
- ▼ We'll excuse Jude for saying it's "child's play" to use because, well, he's right



Do you know HOW ROBOTS SEE?



Buy in the USA:



PiShop.us/stereopi

Buy in Canada:



BuyaPi.ca/stereopi



RASPBERRY PI INSIDE



STOCK RASPBIAN
SUPPORT



OPEN SOURCE



CROWDFUNDED
PROJECT

StereoPi is an open source stereoscopic camera, based on Raspberry Pi.

This is the friendliest way to play with the stereoscopic vision for the spatially aware robots, 3D photo and video!

LinuxGizmos.com

"The StereoPi can capture, save, livestream, and process real-time stereoscopic video and images for robotics, AR/VR, computer vision, drone instrumentation, and panoramic video."

MickMake

"With it you can do things like, stream stereoscopic 3D video to YouTube, build real-time depth maps using OpenCV, create panoramics using Hugin and even a 3rd person view of real life. Cool."

Raspberry Pi Blog

"There are some excellent community efforts too, of which our current favourite is this nifty dual camera board."

Hackster News

"You can hook this up to YouTube, to Oculus Go, you can use it with OpenCV.. I cannot wait to start messing around with these because it's basically a dream come true."

Retro gaming NES frame

Ashley Green doesn't need to fire up his NES console to enjoy a pang of nostalgia, thanks to his cart-based retro art frame, as **David Crookes** discovers



Ashley Green

Ashley is an instructional learning designer in London and a self-confessed entry-level geek who's also spent time as a graphic designer and video editor. In his spare time he enjoys looking for creative solutions to non-existent problems.

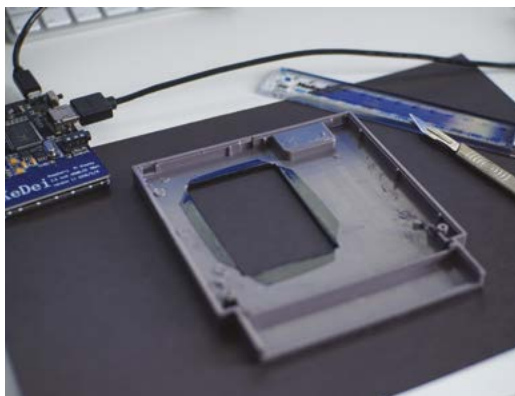
nostalgiaop.app

Pixel art was once a means to an end: a way of creating graphics in the early days of computers and consoles when visual capabilities were far less advanced than today.

It combined tiny blocks of colour, often using a limited palette, and you only need to look at games from the 8-bit and 16-bit era to see many striking examples.

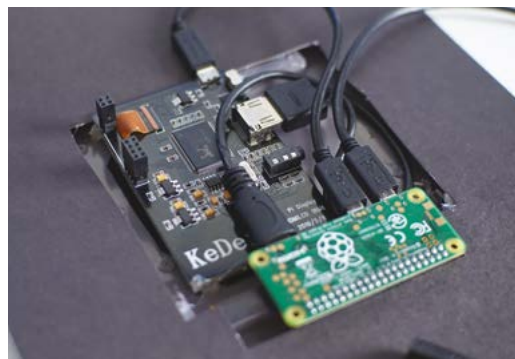
Today, pixel art invokes pangs of nostalgia and it has evolved into a much-loved digital aesthetic in its own right. Ashley Green is certainly a huge fan, which is why he sought a quirky way of displaying his favourite pixel art, hitting on an idea that also paid homage to the past.

"I'm the proud owner of a few Raspberry Pi computers and I'm always looking for new and interesting ways of using them, with retro gaming being the main driver," he says. "Having previously used Raspberry Pi 3 in a mini-replica NES case, I always thought something was missing: the classic NES cartridge itself. There's something extremely satisfying about handling physical media that you don't quite get with emulation."



▲ Be careful: the case will need to be cut. Ashley has also used black vinyl for a finishing touch

By opting for the WH model of Raspberry Pi Zero, the 40-pin GPIO header is professionally pre-soldered, allowing for a simple connection with the display



▲ As well as connecting via GPIO, Raspberry Pi Zero can be simply hooked up using cables

Playing with power

Rather than amass a large collection of cartridges – "I didn't want to break the bank or shelves with stacks of old games" – Ashley felt that a single cart could be used to showcase the style of art he loves.

Inspired by a project that turned an NES case into a full handheld system complete with screen and controller, he decided to turn a cartridge into an art frame which could then be placed in his study as a constant reminder of the past.

"I thought it would allow me to appreciate the time and effort that went into the artwork and give me a little preview of classic NES games," he reveals.

Let's-a-go!

Grabbing a Raspberry Pi Zero W and a 3.5-inch screen that he was using for another project, he got to work. "Not wanting to destroy an original cart, I searched online and found a new blank cartridge shell. All I then needed was the USB and HDMI cables as well as a slim battery pack."

Since the screen is placed in the area usually reserved for the cartridge label, the build required



Quick FACTS

- ▶ No original carts were harmed
- ▶ Connect the screen via GPIO for a slim fit
- ▶ The battery packs last for about six hours
- ▶ The media can be streamed via WiFi
- ▶ Yodeck is free for single-screen use

“Pixel art invokes pangs of nostalgia and it has evolved into a much-loved digital aesthetic”

some cutting into the plastic. Most of the project, however, involved setting up the software, with Ashley using the cloud-based digital signage app Yodeck. This allowed him to create, schedule, and upload videos and images from his computer or smartphone, although he says OXMPPlayer would work well for offline creations.

“I found the best process was to upload media and create a playlist that I could place on a layout that suited the screen,” he says. “I’ve chosen to use mine to show a mix of artwork, game footage, and classic NES commercials, but I’ve also used it to show off an action RPG called Induction Quest, which I created in Unity. It certainly makes for a great conversation piece – I’ve had a few people ask me what it is during my work video calls.”



▲ Ashley has framed his cartridge, shown here running footage from his own game, Induction Quest. He's currently working on SNES and VHS cassette versions



Atomic TV

Nothing on television worth watching? Ryan Cochran's TV set is just as visually arresting when it's turned off, as **David Crookes** reports



Ryan Cochran

Ryan owns Atomic Industry. A media company running websites about vintage hot rods, tools, garage organisation, and more. He enjoys woodworking and wrenching on old cars.

magpi.cc/atomictv

Flat-screen televisions, with their increasingly thin bezels, are designed to put the picture front and centre. Go back a few decades, however, and a number of TVs were made to look futuristic – some even sported space age designs resembling astronaut helmets or flying saucers sat upon elaborate stands. They were quirky and hugely fun.

Maker Ryan Cochran's project evokes such memories of the past. "I have a passion for vintage modern design and early NASA aesthetics, and I wanted to make something which would merge the two into an art piece that could fit on my shelf," he recalls. "The first thing I could think of was a small television." And so the idea for the Atomic TV came into being.

Made of wood and using spare tech parts left over from a couple of past projects, it's a television that's as compelling to look at when it's turned off as when it's playing videos on a loop.

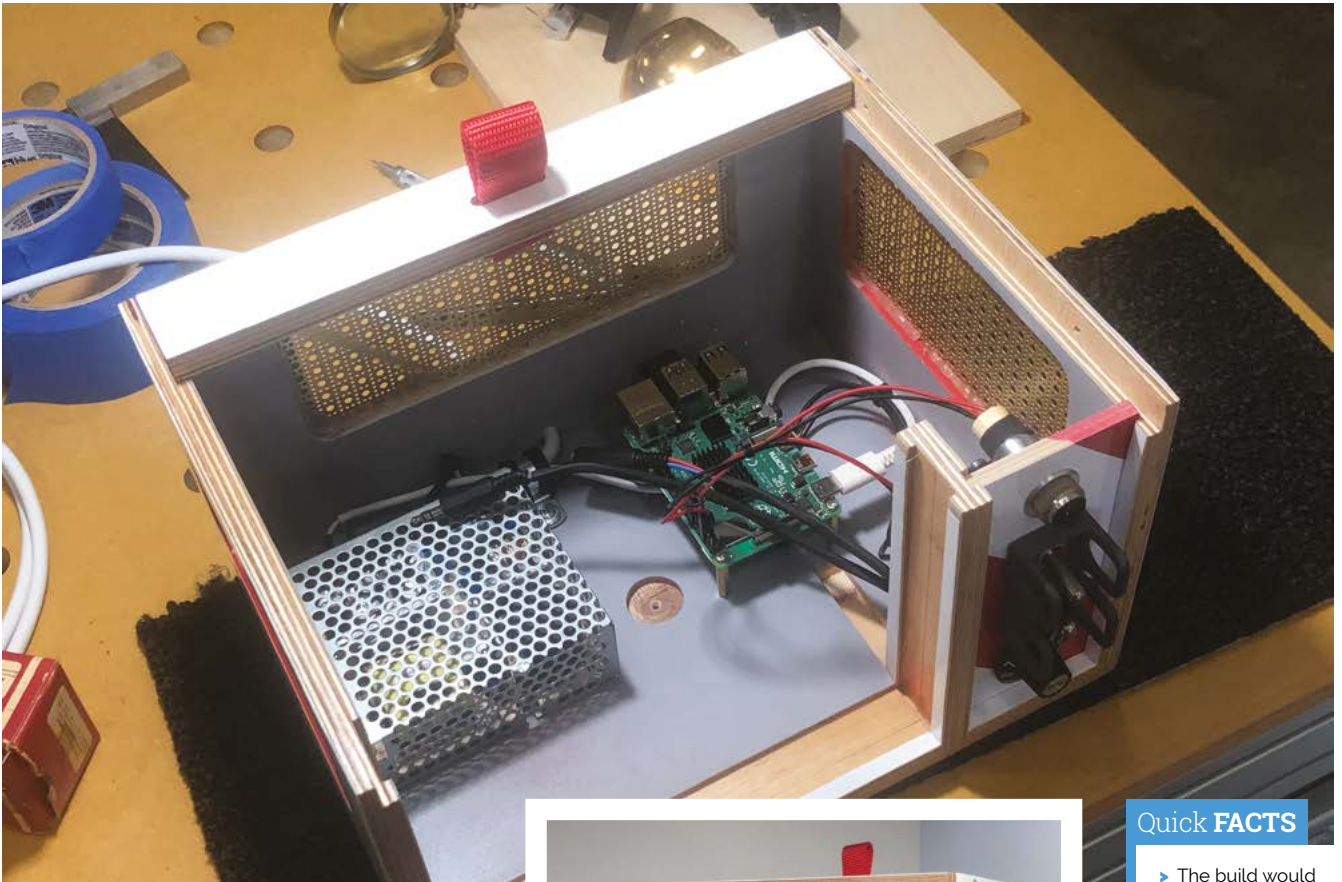
"My main concern was fit and finish," he says. "I didn't want this thing to look amateurish at all. I wanted it to look like a professionally built prototype from 1968."

Turn on

Before he began planning the look of the project, Ryan wanted to make sure everything would connect. "The parts sort of drove the direction of the project, so the first thing I did was mock everything up without a cabinet to make sure everything worked together," he says.

This posed some problems. "The display is 12 volts, and I would have preferred to simplify things by using one of the 5-volt displays on the market, but I had what I had, so I figured a way to make it work," Ryan explains, discovering the existence of a dual 5V-12V power supply.

With a Raspberry Pi 4 computer, the LCD display, a driver board, and a pair of USB speakers borrowed



▲ A Raspberry Pi 4 sits neatly inside the project's bespoke housing, secured using 3mm standoffs

from his son all firmly in hand, he worked on a way of controlling the volume and connected everything up.

"Power comes in and goes to an on/off switch," he begins. "From there, it goes to the dual voltage power supply with the 12V running the display and the 5V running Raspberry Pi 4 and the small amp for the speakers. Raspberry Pi runs Adafruit's Video Looper script and pulls videos from a USB thumb drive. It's really simple, and there are no physical controls other than on/off switch and volume."

“The television runs 24/7 and it spends 99 percent of its time on mute”

Tune in

The bulk of the work came with the making of the project's housing. "I wanted to nod the cap to Tom Sachs, an artist who does a lot of work I admire and my main concern was fit and finish," Ryan reveals.

He filmed the process from start to end, showing the intricate work involved, including a base



▲ It's turned off now, but the 1940s-era Stewart Warner Indicator power light shines when electricity is flowing to the power supply

created from a cake-stand and a red-and-white panel for the controls. To ensure the components wouldn't overheat, a fan was also included.

"The television runs 24/7 and it spends 99 percent of its time on mute," says Ryan. "It's literally just moving art that sits on my shelf playing my favourite films and video clips and, every now and then, I'll look over, notice a scene I love, and turn up the volume to watch for a few minutes. It's a great way to relax your brain and escape reality every now and then." **M**

Quick FACTS

- ▶ The build would cost less than \$200
- ▶ Ryan has 256GB worth of media to play
- ▶ These are loaded on to a thumb drive
- ▶ The videos are coded to automatically loop
- ▶ This was Ryan's first project using Raspberry Pi

Oktoberfest Pinball Machine

Real-life arcade halls may be off-limits, but by the power of Raspberry Pi you can play real pinball online. **Rosie Hattersley** flips out



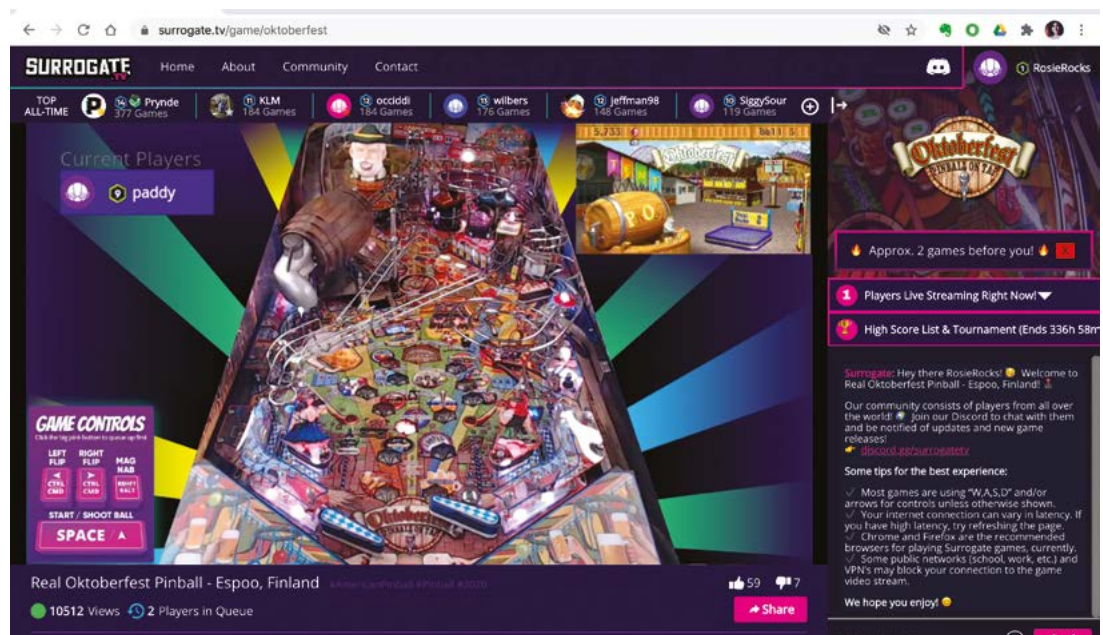
Stan Dmitriev

Stan works at real-life online games company Surrogate.tv in Espoo, Finland. Their marvellously geeky office is strewn with more than 100 Raspberry Pi 4 computers.

surrogate.tv

Many of us dream of having our own games room, quite possibly with a pinball machine at its centre. Such fantasy comes with a rather large price tag, so the idea of being able to play a proper pinball machine, albeit via the internet, holds a strong appeal. This was the thinking behind Finnish company Surrogate.tv, which specialises in creating real-life games experiences that can be played online. The two-year-old company has recently launched its first Raspberry Pi-based pinball game, the Oktoberfest Pinball Machine, in which players shoot balls to gradually fill a stein of beer while an oompah band sings a toast.

While the theme may seem timely since the legendary Munich beer festival ought to be taking place in September, Surrogate.tv's Stan Dmitriev tells us that it just so happened this design of vintage pinball machine came up for sale while they were pondering a Raspberry Pi 4-based design. The 20-strong staff of Surrogate.tv are "a bunch of gamers and geeks," says Stan. He estimates that there are more than 100 Raspberry Pi 4 computers in the office! Some have been used to control various robots – one of their other offerings is a SumoBot challenge in which real sumo robots battle it out in an arena, controlled



► Play the real Oktoberfest Pinball Machine for free in Chrome or Firefox



SURROGATE TV

A Logitech webcam trained on the pinball machine instantly relays the action to the online player

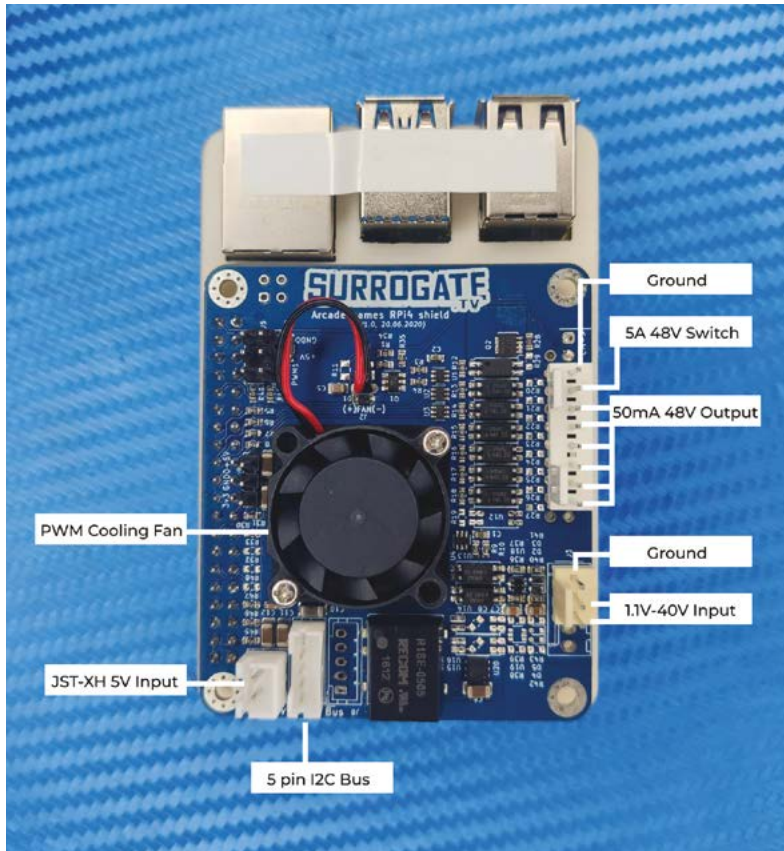
Switches on the specially designed Surrogate Raspberry Pi HAT start a game and activate the pinball plunger and flippers

The controls of an original pinball machine have been reconfigured so it can be played over the internet

Quick FACTS

- ▶ The oompah-accompanied Oktoberfest Pinball Machine runs 24/7
- ▶ Thankfully, it lives in a sound-insulating cabinet
- ▶ On May The 4th Day, Surrogate.tv launched a *Star Wars* special.
- ▶ „Raspberry Pi-powered The Force Claw
- ▶ Some multi-Raspberry Pi 4 projects are in the pipeline

- ▶ Getting Raspberry Pi to recognise the start and end of a game centred on how the Start button's LED behaved



- ▲ Surrogate.tv designed a Raspberry Pi 4 arcade game HAT for use with different types of arcade machine



**Alert!
High voltage**

Be careful when working with mains electricity.

magpi.cc/electricalsafety

by players over the internet. However, they'd also have a powerful computer on the side to execute the ultra-low-latency video streaming part of the project stack.

Flipping fun

The Oktoberfest Pinball Machine is the first project they've created in which Raspberry Pi handles the whole setup. "Players can control the flippers using the left and right **CTRL/CMD** key and can shoot the ball from the plunger using the **SPACE** bar on the keyboard," explains Stan. Webcams provide a live feed of the pinball bouncing around inside the machine, with the human player responding by hitting the flipper buttons to keep it moving and away from the danger zone. One webcam shows the action, while another shows the current and high scores. "Both

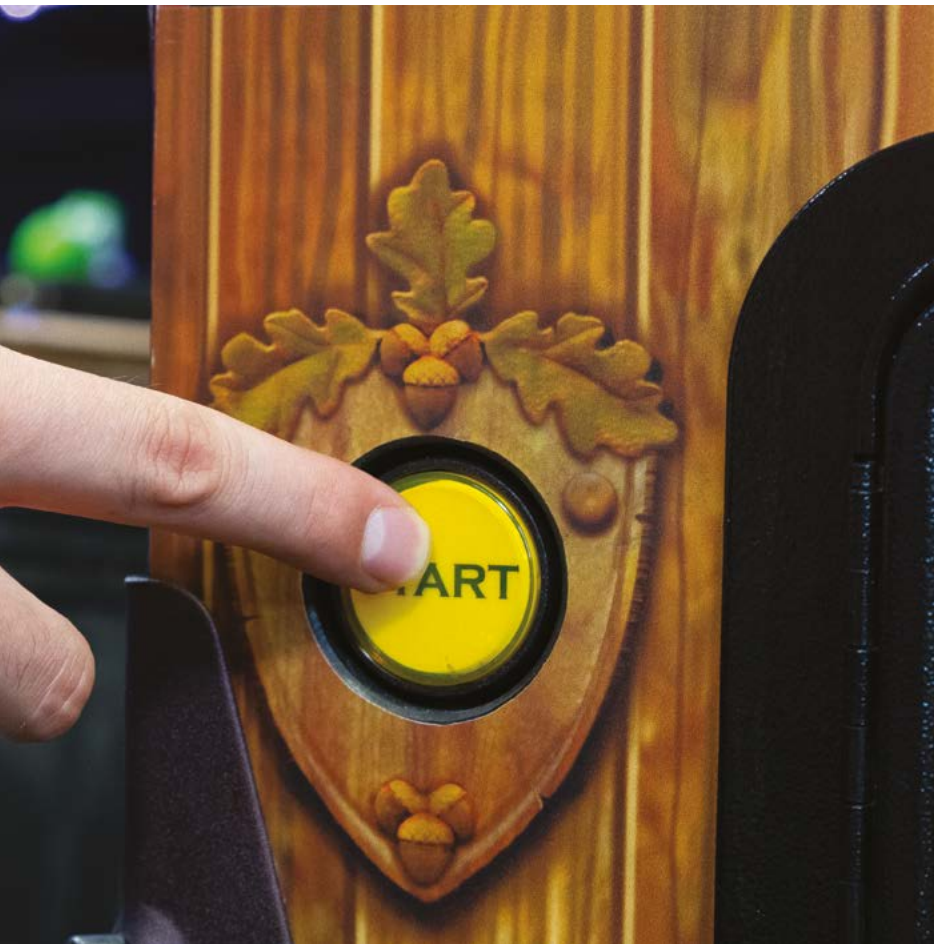


these video feeds and the audio feed are combined using Raspberry Pi's compute capabilities, encoded into a single compressed video feed and sent to the user in near real-time using the SurroRTG suite," he says.

Like in a real arcade hall, you have to queue to play, since you really are playing the actual machine. Blinking LEDs on the machine at the end of a game are recognised by Raspberry Pi, which then invites the next person to play. They undertook lots of analysis before realising the Start button LED operates in a very specific way based on the pinball machine's state, says Stan. "By reading this behaviour and inputting it to Raspberry Pi, we managed to solve the challenge."

Pinball wizardry

Their existing online pinball machine, Batman '66, proved very popular, and players were asking for more machines. For the Oktoberfest machine, they took a more streamlined approach than the Batman machine, with plans to simplify the process of hooking up pinball machines for online play. "We wanted to make the setup replicable," reveals Stan.



To this end, they designed their own arcade machine HAT with eight output switches. This Raspberry Pi 4 HAT works with a range of arcade machines, including claw ones. It has its own cooling fan, as well as dedicated switches corresponding to arcade machine controls

“ Like in a real arcade hall, you have to queue to play, since you really are playing the actual machine ”

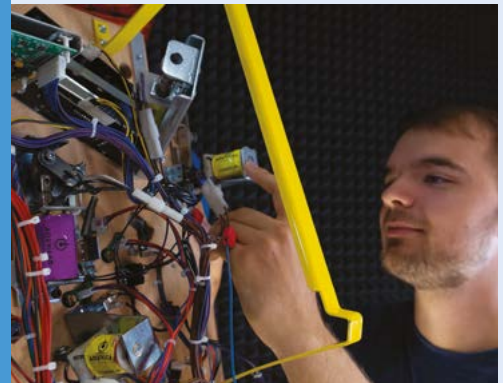
and audio and video feeds. The Oktoberfest machine also uses Surrogate’s own SurroRTG low-latency video codec to enable a smooth gaming experience.

Most of the code to interact with the arcade machine and for the game logic is Python-based. Although the SurroRTG software is not currently available on GitHub, Surrogate.tv are set to launch it as a free download in the coming weeks and reassure us there are exciting plans afoot to entice makers with an eye on creating their own gaming setup. [M](#)

Single-player mode



- 01** To turn an arcade machine into one playable online, download the SurroRTG software (it will be a free download launching this autumn), install it on Raspberry Pi 4, and navigate to surrogate.tv.



- 02** Hook up the wires for the flipper and control buttons on your arcade machine to corresponding pins on your Raspberry Pi, then wire up the Start button LED and the ball lane detector sensor.



- 03** Position webcams so the pinball machine screen can be seen, then attach them via USB to Raspberry Pi 4. Either connect the machine's audio to a USB audio card or use the webcam microphone. Plug in an Ethernet cable and power source.



Retro TV MakeCode Arcade Machine



Alan Wang

Based in Taiwan, Alan is a former book translator and currently a junior editor in a local programming book publisher. He makes stuff as a hobby.

magpi.cc/retrovarcade

If building a full-size arcade cabinet is a little too daunting, why not create a mini one using a Raspberry Pi Zero?, asks **Nicola King**

When Alan Wang started tinkering with Microsoft's MakeCode Arcade editor, it sparked an idea and he decided to make a tangible and authentic-looking arcade machine to achieve a fully nostalgic gaming experience.

"I wanted to make a physical arcade machine with an Adafruit Metro M4," he recalls. "However, [Microsoft MakeCode] does not have a firmware for this exact board... I didn't know how to do soldering back then, so I didn't pick ItsyBitsy or Feather M4." Adafruit projects are almost unheard of in Taiwan, where Alan lives.

"Later I found out there's a firmware for Raspberry Pi Zero, which I happened to have one

[of]. Compared to microcontrollers, the firmware for Raspberry Pi Zero is surprisingly easy to install." Add in a small HDMI LCD screen, some cables, and a button module for game controls, and Alan was nearly there.

Tiny controls

Not too complicated to put together, Alan explains that once the MakeCode Arcade firmware is loaded, Raspberry Pi Zero will boot to the Arcade system. "Quick and easy. Buttons need to be connected to Raspberry Pi Zero." For this, he connected a button module with a tiny joystick, adding an extra push-button. "I wish I knew how to make stuff with 3D printing or laser cutting, otherwise I would make my own arcade controller too."

It proved a very quick make for Alan. He says most of the time spent was in trying to fit the Raspberry Pi Zero and LCD into the tiny vintage-TV-style wooden housing, and the only coding

" I wish I knew how to make stuff with 3D printing or laser cutting, otherwise I would make my own arcade controller too **"**



▲ A 4-inch LCD is connected via HDMI, but enlarged by the magnifying plate

required involved modifying the **config.txt** file in the Raspberry Pi firmware to make the LCD screen work.

Screen magnifier

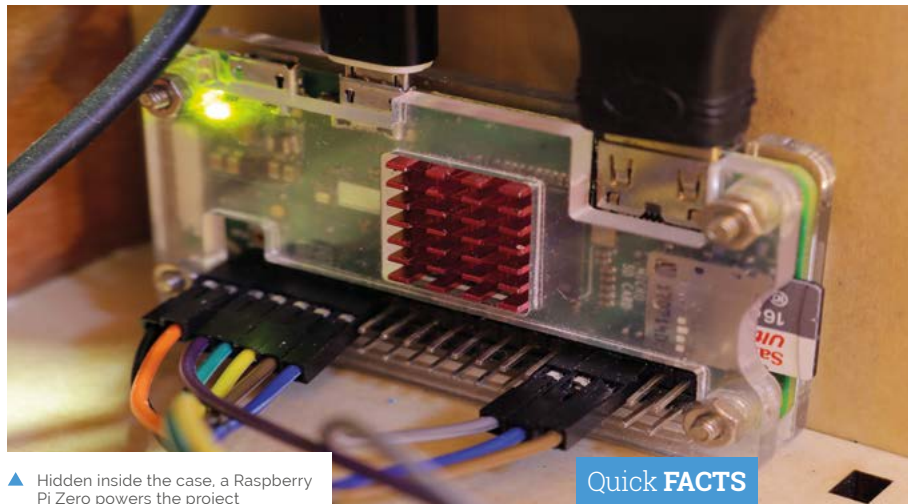
The physical appearance of the mini machine was important in order to really capture that crucial retro feel, so Alan chose to house the inner workings inside a WOODSUM 3D Puzzle Smartphone TV. Originally designed for use with smartphones, this case magnifies the screen for watching videos and so on.

Alan immediately recognised the possibilities, as he shares: “Since small HDMI LCDs for Raspberry Pi [computers] are more or less the same size as smartphone screens, in theory I can use them in the TV box as well, right?”

What’s more, the effect achieved using this particular case added to the vintage feel. “The magnifying plate of the TV box would enlarge and slightly distort the LCD. It would look a bit like an old cathode-ray tube TV,” explains Alan.

Now, you may be wondering how easy it is to play games on such a tiny screen, but Alan says he’s found it pretty simple. “MakeCode Arcade is designed for devices with 160×120 TFT displays, which are smaller than my 800×480 LCD. And the magnifying plate makes it look like four times bigger.”

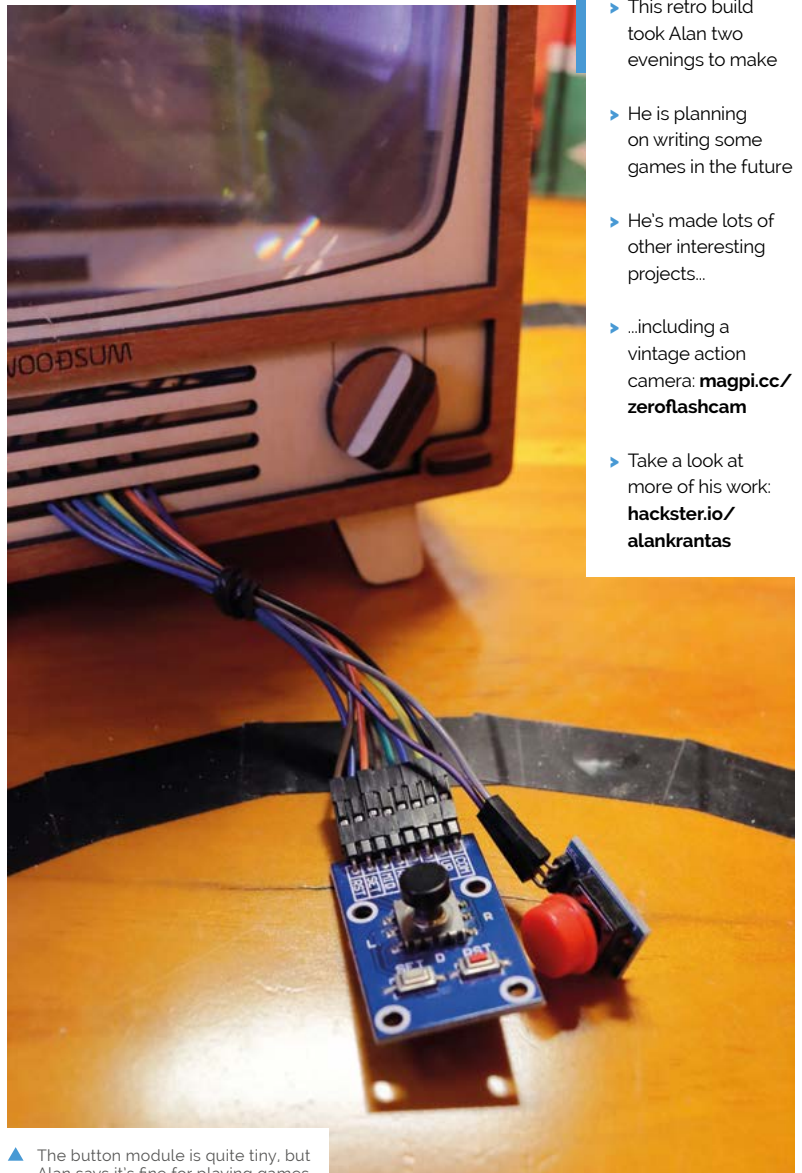
All in all, Alan has found his make very reliable so far, although he says some games downloaded from the MakeCode Arcade website may crash it: “It’s software compatibility issues I guess; I can avoid them if I am writing and testing my own games.” And, surely a lot of the fun is in writing and testing your own games – why not give it a go? [M](#)



▲ Hidden inside the case, a Raspberry Pi Zero powers the project

Quick FACTS

- ▶ This retro build took Alan two evenings to make
- ▶ He is planning on writing some games in the future
- ▶ He’s made lots of other interesting projects...
- ▶ ...including a vintage action camera: magpi.cc/zeroflashcam
- ▶ Take a look at more of his work: hackster.io/alankrantas



▲ The button module is quite tiny, but Alan says it’s fine for playing games

Black Lives Matter projection



Kareem Rahma

Egyptian-American Kareem is a NY-based artist and media entrepreneur. He co-founded The Nameless Network and The Museum of Pizza and previously worked at VICE.

@kareemy



David Dellanave

Entrepreneur David is a coach, gym owner, and olive oil importer who has been deeply involved in open-source technologies since the late 1990s.

dellanave.com/
oliveoil

A Raspberry Pi-controlled video projection proclaims the rights of Black people. **Rosie Hattersley** gets the message

Before daybreak one morning in June, a huge 120×100 ft (30.5×36.5 m) video projection appeared on the wall of the Gold Medal Flour Mills building in Minneapolis. Gil Scott Heron's iconic track *The Revolution Will Not Be Televised* played over a looped set of images and video clips featuring US civil rights leaders. The installation was a powerful message of how things must and will change. The thought-provoking projection is powered, of course, by Raspberry Pi.

Its creators, artist Kareem Rahma and entrepreneur David Dellanave, were only too happy to fill *The MagPi* in on how the project came about.

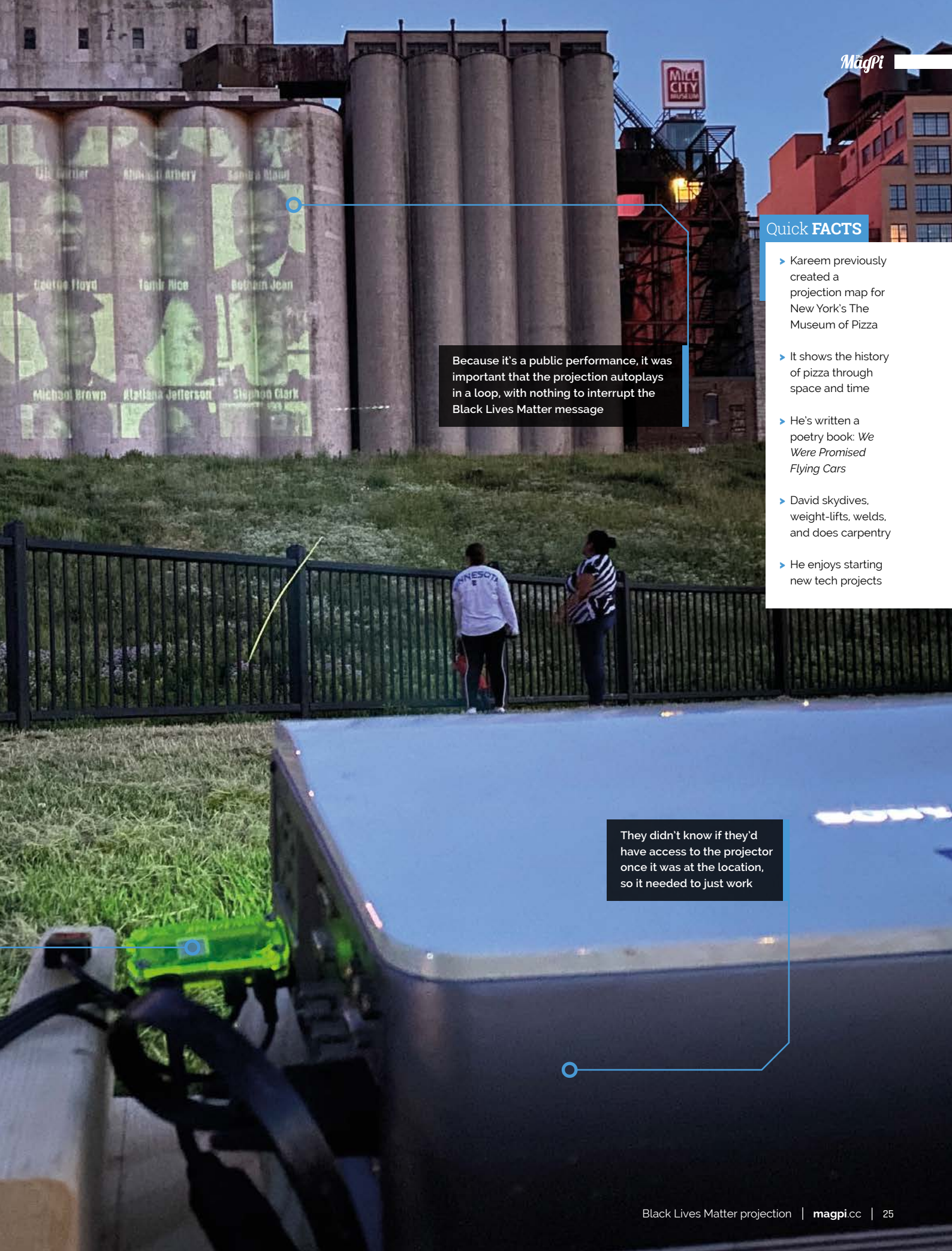
Historical significance

Kareem explains: "The idea was to display a juxtaposition of peaceful protestors with violent, oppressive police and elevating the names of Black people who have been killed by police in America." Having had the germ of an idea for such a project, New York-based Kareem tweeted asking for someone to help him make it happen. Minnesotan David, who he'd worked with a decade previously, responded. The two worked remotely until the time came to install their project at its location.

David suggested making use of Raspberry Pi. He describes it as his "go-to for small projects and display dashboards" at the gyms he administers, rather than politically charged public statements. "I've never done anything outside the realm of the strictly practical, so this was a cool opportunity to be a part of," he says.

Raspberry Pi Zero auto-mounts the USB drive and plays images of important civil rights campaigners and video clips of historic speeches stored on it





Quick FACTS

- ▶ Kareem previously created a projection map for New York's The Museum of Pizza
- ▶ It shows the history of pizza through space and time
- ▶ He's written a poetry book: *We Were Promised Flying Cars*
- ▶ David skydives, weight-lifts, welds, and does carpentry
- ▶ He enjoys starting new tech projects

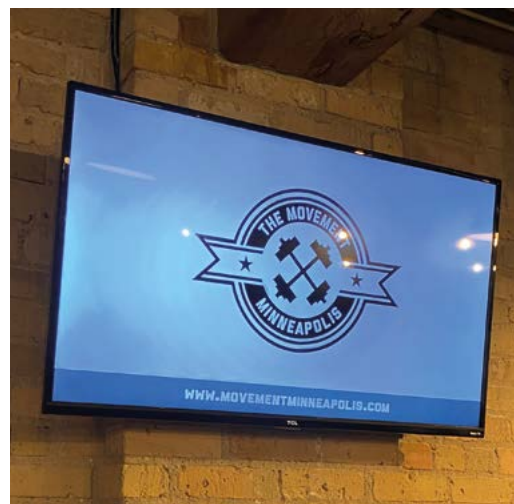
Because it's a public performance, it was important that the projection autoplays in a loop, with nothing to interrupt the Black Lives Matter message

They didn't know if they'd have access to the projector once it was at the location, so it needed to just work



▶ The projection featuring iconic civil rights campaigners, such as Martin Luther King, can be seen across Minneapolis

▶ Raspberry Pi Zero's low cost appealed partly because the setup might get confiscated or destroyed



▶ David usually uses Raspberry Pi for displays at his gym



Due to the nature of the project, there was always a risk that “the setup could get confiscated or destroyed,” says David. “So it was a huge advantage to have hardware that was cheap enough to be practically disposable.”

Kareem is also a newcomer to public art displays, though he did help produce a projection mapping project for The Museum of Pizza in New York!

For the Black Lives Matter projection, Kareem says: “We wanted to do something that would shock people a little bit, make them uncomfortable, and take advantage of this period of calm that was happening at night-time due to politicians successfully suppressing protests.” They worked with producer Hayley Pappas, creative director Smiley Stevens, and editor Khalil Anderson in Los Angeles to produce the first set of images and videos. Subsequent sets of content



for the display were suggested by the local community in Minneapolis, as well as including transcribed speeches by Malcom X, James Baldwin, and leaders.

▲ Kareem previously helped create the story of pizza through time for a dedicated Museum of Pizza. Only in New York, right?

“ With protests in cities across the US resulting in government-imposed 10pm curfews, their projection was a risk ”

Long-distance protest piece

Minnesota-based David already had all the parts the project required. “I chose the non-wireless Raspberry Pi Zero specifically so there were no remote attack vectors possible. It’s more than powerful enough to play the content with no issue.”

He built a small tiltable platform for the projector from wood scraps in his garage. The setup runs off a small generator, so it’s mobile and not dependent on a power outlet.

The software side worked perfectly from the first go, he says. The advice of other Raspberry Pi and commodity hardware users “cannot be overstated” says David, who admits he’s not the most talented coder or software engineer.

Any tweaking needed was to get the optimum projection location so it covered the entire Gold Medal Flour building. “The site is historic, iconic, and massive. It’s also central to Minneapolis and can be seen from different parts of the city,” explains Kareem. With protests in cities across the US resulting in government imposed 10pm curfews, their projection was a risk.

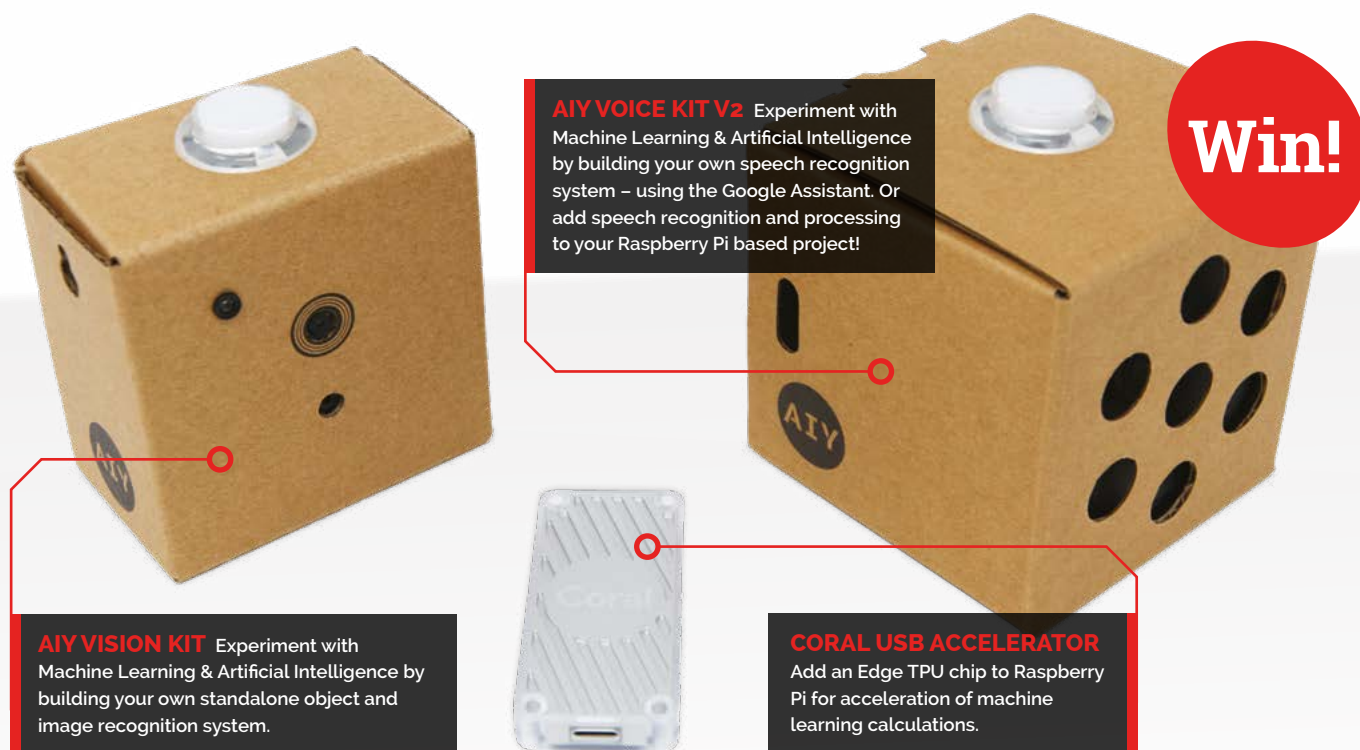
Responses to the projection were immediate and very enthusiastic. “We’re very happy with how the idea has been received and hope that our project has inspired others,” says Kareem. “We’d love to see others replicate the idea, especially with permanent installations!”



Warning! Public projection

Be mindful of local laws regarding public screening in your area. Seek permission from the owner of the wall or projection surface, or provide your own screen. Adhere to public safety regulations. And be mindful of broadcasting regulations in your area. You need a ‘non-theatrical’ film licence to show films and TV programmes in public in the UK.

magpi.cc/projectionlaw
magpi.cc/showingfilms
magpi.cc/filmlicence



Add AI to your project & pi3g will supply the kit

Tell us what you'd like to make! Then pi3g will send five makers an AIY Voice Kit v2, AIY Vision Kit, or Coral USB Accelerator. Plus! Your project can appear in The MagPi

Coral



A IY Projects and Coral are powerful AI (artificial intelligence) kits you can use with your Raspberry Pi.

All three different products are now available via pi3g (an official distributor for Google Coral & AIY Projects) and the retailers shown on the opposite page.

pi3g plans to celebrate AIY Projects and Coral by giving away five AIY Voice, AIY Vision, or USB Accelerator units to *The MagPi* readers. And we will make sure the world gets to see your creation!

All you have to do is tell us what you plan to make with your AIY Voice Kit, AIY Vision Kit, or USB Accelerator. If your project is selected, pi3g will send you the kit you need and **you will appear in a future edition of *The MagPi* magazine.**

The **AIY Voice Kit v2** enables you to explore voice recognition and natural language.

Using **AIY Vision Kit** you build an intelligent camera that identifies objects.

Both AIY Voice v2 and AIY Vision kits include a Raspberry Pi Zero WH and the cardboard DIY cases (for easy assembly and no soldering required!).

The **Coral USB Accelerator** brings the Edge TPU (Tensor Processing Unit) to Raspberry Pi. TPU is Google's purpose-built chip designed to run AI at the edge: your projects are capable of performing fast and efficient AI calculations. Add features such as real-time object recognition and classification, speech processing, and smart decision making. All while maintaining data privacy! It goes great with Raspberry Pi 4's USB 3.0 port.

All three devices are now available, and you can pick them up from the retailers listed opposite.

So don't delay! Let us know today what you could do with artificial intelligence. magpi.cc/pi3g

Enter now: magpi.cc/pi3g

Get inspired with amazing AI projects



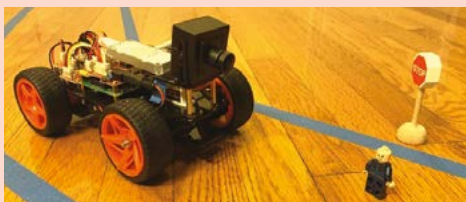
▲ Smart Bird Feeder. Follow the instructions at Coral's website to build a smart birdhouse that records birds and deters squirrels (magpi.cc/smartbird)



▲ Martin Mander's Smart Intercom upcycles a retro machine with AIY Voice (magpi.cc/aiyintercom)



▲ Alasdair Allan's AIY Magic Mirror uses both AIY Voice and AIY Vision kits to recognise your face and respond to spoken commands (magpi.cc/aiymagicmirror)

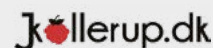
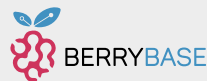


▲ David Tian has turned an off-the-shelf Pi-Car V robot into self-driving car using a Coral USB Accelerator (magpi.cc/deeppicar)



▲ These dinosaur toys have been turned into Do-you-think-he-saurs. Inside both are AIY Vision kits. Now their faces move around and track people (magpi.cc/aiytoy)

Check these stockists to buy your AIY Projects and Coral kit

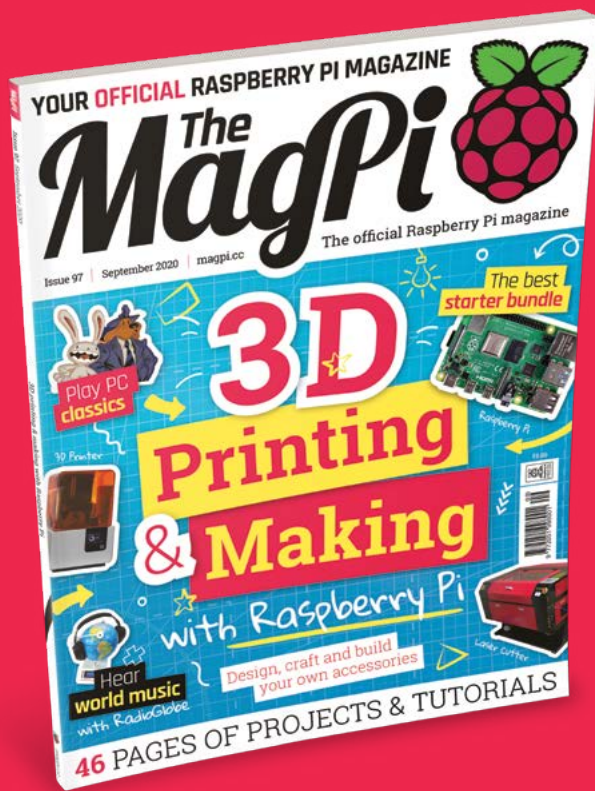


Add artificial intelligence to your Raspberry Pi project today! Buy an AIY Voice Kit, AIY Vision Kit, or Coral USB Accelerator and dive deep into the amazing world of AI. The most exciting and important area of computing to discover!

pi3g.com/coral

SUBSCRIBE TODAY FROM ONLY £5

SAVE UP TO 35%



Subscriber Benefits

- ▶ **FREE Delivery**
Get it fast and for FREE
- ▶ **Exclusive Offers**
Great gifts, offers, and discounts
- ▶ **Great Savings**
Save up to 35% compared to stores

Rolling Monthly Subscription

- ▶ Low monthly cost (from £5)
- ▶ Cancel at any time
- ▶ Free delivery to your door
- ▶ Available worldwide

Subscribe for 12 Months

£55 (UK)	£90 (USA)
£80 (EU)	£90 (Rest of World)

Free Raspberry Pi Zero W Kit with 12 Month upfront subscription only (no Raspberry Pi Zero W Kit with Rolling Monthly Subscription)

📞 Subscribe by phone: **01293 312193**

📧 Subscribe online: **magpi.cc/subscribe**

✉ Email: **magpi@subscriptionhelpline.co.uk**

JOIN FOR 12 MONTHS AND GET A

FREE Raspberry Pi Zero W Starter Kit

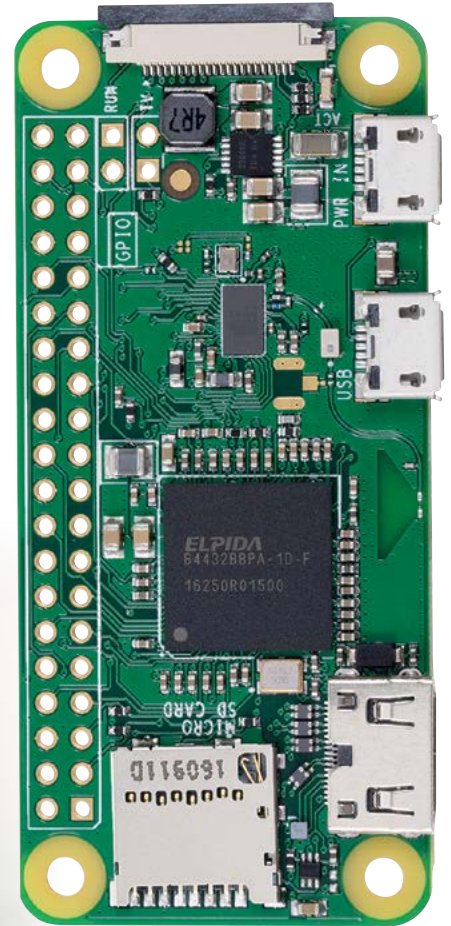
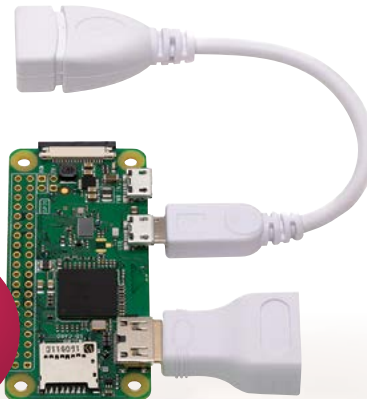
WITH YOUR FIRST
12-MONTH SUBSCRIPTION

Subscribe in print
today and you'll
receive all this:

- ▶ Raspberry Pi Zero W
- ▶ Raspberry Pi Zero W case with three covers
- ▶ USB and HDMI converter cables
- ▶ Camera Module connector

This is a limited offer. Not included with renewals. Offer subject to change or withdrawal at any time.

WORTH
£20



 Buy now: magpi.cc/subscribe

SUBSCRIBE on app stores

From **£2.29**

 Available on the
App Store

 GET IT ON
Google Play

3D Printing & Making with Raspberry Pi

Craft your own
accessories from
start to finish

By PJ Evans &
Tim Richardson

Raspberry Pi has brought a lot of people to the maker community. All it takes is imagination. Skill is not required: you will learn along the way! Many of the notable names in the Raspberry Pi community came to it with little or no maker skills and it was through experimentation, determination, repetitive failure, and lots of practice that they became the inventive lot they are today. Here we take a look at one of those people and the tools they use for making. Get inspired and you too could be in these pages one day.

MAGDOWNLOAD.ORG

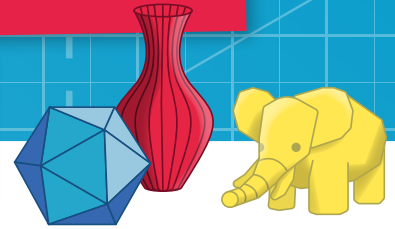
FREE DOWNLOAD

LATEST MAGAZINES

HIGH QUALITY TRUE-PDF

MAGDOWNLOAD.ORG

3D printing & more & Raspberry Pi



Maker

Tim Richardson

Tim has been involved with the Raspberry Pi community from almost the start. He is part of the Pi Wars organising team and a course designer/builder, as well as writing the CamJam EduKit worksheets, CamJam organiser, and now... PCB designer.

@Geeky_Tim



Visit OctoPrint

Install OctoPrint on Raspberry Pi OS, or download the OctoPi OS from octoprint.org.

How to get started with 3D printing, laser cutting, and PCB design

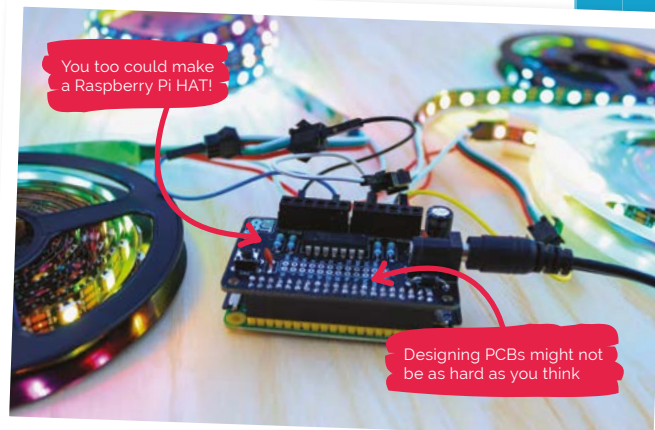
After seeing his first 3D printer at a Raspberry Jam back in 2014, **Tim Richardson** bought one. They were just starting to become affordable, albeit £600 for a 'budget' one back then! They are much more affordable now, as little as £150–£200 for a decent one. He has some advice for those new to or thinking about getting one.

OctoPrint

Tim suggests that one of the best upgrades you can do is to add a Raspberry Pi computer running OctoPrint (octoprint.org). It's free and open-source and has been continuously developed by Gina Häußge since 2012. OctoPrint is used to control and monitor your printer, even remotely, and uses a Raspberry Pi Camera Module for creating a time-lapse video of your prints.

OctoPrint runs on almost any Raspberry Pi computer, but you will get the best UI response from a Raspberry Pi 3 or newer.

OctoPrint supports most consumer printers on the market, so it is likely to work with yours. Installation is a breeze! Just download OctoPi, an OS image with OctoPrint pre-installed, write it to a microSD card, boot up, connect your printer, and that's it!



Once you have set up OctoPrint for your printer, you can start moving the print head around (essential for bed levelling) and see the temperature of your print bed (if it is heated) and the extruder.

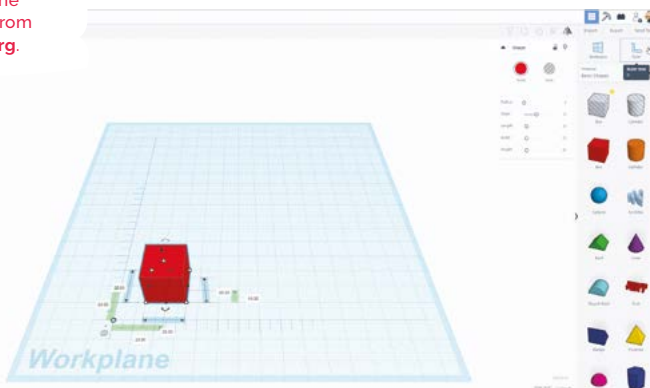
During printing you can watch the temperatures, see the G-code as it is executed, and watch the progress of the print on your phone or computer.

Tinkercad

Tim uses Tinkercad (tinkercad.com) for most of his designs; it's simple and easy to use. He has a few tips which may help you design objects in this online tool from Autodesk.

On starting a new design, add the ruler to the workplane. Every selected shape will show dimensions which can be edited with exact sizes. This even works for rotation.

When designing a mount or case for something, Tim first models the item itself, simplified but accurately measured. He places solid blocks where ports or buttons need to be accessed. He then enlarges the model by 1 mm in each direction. After that, he changes the shape to be a 'hole' and uses it to remove material from simple blocks.



▲ Use the ruler tool to enable precise sizing of objects in the workplane



▲ The Beast in Hut 8!

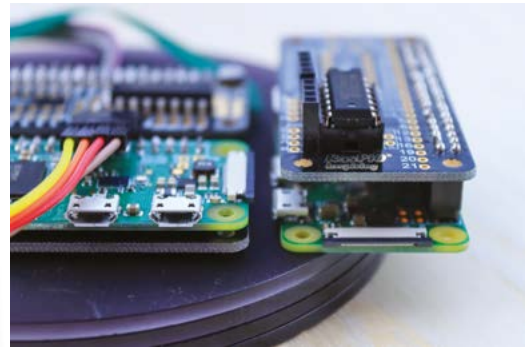
Light work

Laser cutters are the mainstay of almost all makerspaces. Tim is fortunate enough to have a large one of his own that he uses for building Pi Wars courses, but for most people the smaller cutters are more than enough, and cost around £300. Tim has a few bits of advice if you are using one for the first time.

Laser cutters can be used for plastic, wood, and other materials. Some plastics just melt or burn, while some wood is hard to cut and you may end up breaking Rule Zero (don't be on fire). Only buy materials designated as laser safe.

Before cutting expensive materials, always do a test cut in cardboard first. While the thickness will usually not be right, you will be able to line up all the holes to ensure that everything fits. There is nothing more frustrating than carefully designing something only to find that it won't fit together, or you cannot bolt your Raspberry Pi computer in place because you have forgotten about the space the cables take!

Inkscape is a free and great tool to start with, which of course runs on Raspberry Pi. However, for some cutters the SVG files have to be exported to a different format. Dominic Morrow, from Smoke and Mirrors, recommends using Lightburn (magpi.cc/lightburn). It is commercial software, but updated regularly. It is able to connect directly to many laser cutters and control them, all in one tool. Inkscape does have some specific plug-ins which are very useful, though – especially the 'living hinge' tool for cutting 'bendable' wood and plastic.



▲ Honey, I shrunk the kids!

Doing the impossible

Before Raspberry Pi, Tim knew nothing about electronics. Since Raspberry Pi, though, he has brought out three CamJam EduKits to help others learn. Tim says, "It helped that I knew nothing when writing the worksheets as I had to explain things in a way that noobs would understand."

He thought PCBs were something he could never do, but reading an article in HackSpace magazine gave him an idea. For Tim, the height of a Raspberry Pi Zero with micro-HAT (μHAT) was higher than he wanted; how about mounting a Raspberry Pi Zero and μHAT on the same plane?

He found a μHAT template for KiCad (an open-source tool for designing PCBs) which has a single header and holes in place. With a second header, surely it was a simple case of 'joining the dots'? Not quite – it's advisable to connect all ground pins together. Tim had to move tracks to give space for 'vias' (channels that go between layers of the PCB).

Once confident the connections were correct, it was time for manufacture. Tim searched for PCB makers, but for a simple board he didn't want to pay much. PCBWay (pcbway.com) had an easy-to-use interface and instructions on preparing KiCad designs for manufacture.

PCB design tips

While waiting for the first PCB to be manufactured, Tim started designing the second. He wanted to control WS2812 LEDs (aka NeoPixels) with Raspberry Pi.

When designing a PCB, you first have to find components that do what you want. For WS2812 LEDs, the 3.3V from the

Top tip!

OctoPrint's built-in slicer

OctoPrint can either use G-code from your favourite slicer, or you can slice STL files with the built-in slicer.

GPIO pins must be increased to 5V. The 74HCT125 chip has four 'level shifters' for that.

The next task is to breadboard the circuit, and write code to control the electronics – open-source software can help with this.

Before designing the PCB, you have to design the schematic diagram: how each component connects to other components. It doesn't have to be pretty, but it has to have all the right connections.

Often, specific GPIO pins have to be used. Other pins seem to be logical on the breadboard, but when routing the tracks on the PCB you may find it is not quite as simple! "Be ready to change the design multiple times to make routing of tracks easier," Tim advises. If you have to make any changes during the PCB design, always go back and rework the breadboard.

Eventually Tim got a PCB that looked like it would work. He showed the design to some friends – an invaluable part of designing a PCB. For Tim, they suggested he should break out unused pins and add a button for turning Raspberry Pi off.

After a redesign, Tim asked a PCB expert to check it. They came back with lots of advice.

Firstly, protect the Raspberry Pi computer from powering the LEDs. It cannot supply much current, so a Schottky diode between the PCB's power input and Raspberry Pi will stop that.

There were 'unsightly gaps' on the PCB; the ground plane did not flow into spaces where tracks

were too close. They also advised capacitors to help smooth power supply fluctuations, as well as button 'bounce' for the off button.

Tim says, "Designing PCBs takes multiple iterations, especially if it is your first. Even experts don't get it right the first time."

He ordered this second PCB from PCBWay and, with the express service, had the new boards within a week! However, as often happens, things were not perfect. Tim had placed the silk screen (printing) over components on the bottom of the board. A simple cosmetic mistake, fortunately.

"I soldered the first PCB and... it didn't work," he recalls. "Raspberry Pi Zero did not boot." He

“ Be ready to change the design multiple times to make routing of tracks easier ”

tested the supply and then, with a multimeter, worked through the PCB. The barrel jack had power, but the ground was the wrong leg! The 'footprint' Tim used was not the same as the physical jack.

Fortunately, the fix was simple: soldering a wire between the ground and mounting pins was all that was needed. The rest worked perfectly!


Top tip!

Mind the gap

When building a mount or case, remember you have to be able to slide the device into it! Leave a route in.



Alert! Stay safe

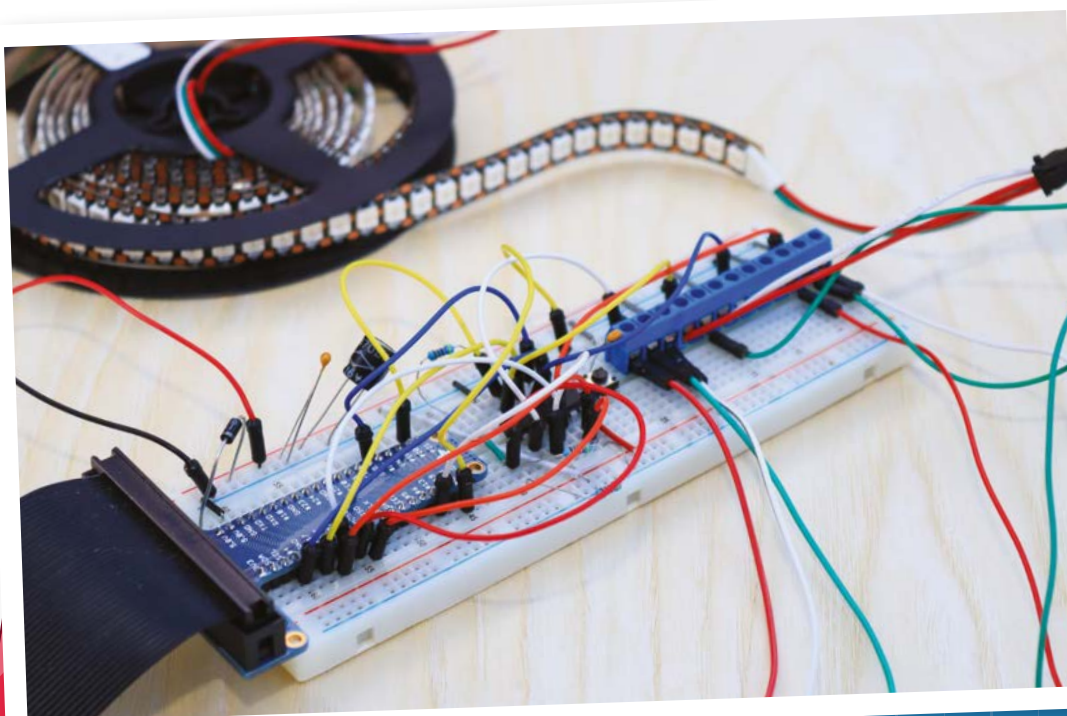
Always look up information on how to use new equipment safely.
magpi.cc/lasercuttersafety
magpi.cc/3dprintersafety

◀ Always breadboard your circuits


Top tip!

And breathe!

Laser cutters work by burning material at high temperatures. Some fumes are dangerous, so always use cutters in a ventilated area.



Design and build an e-ink display stand

Maker

PJ Evans

PJ is a writer and software engineer. His 3D printer was once described as a 'device for making Christmas cracker tat' and he's not having it.

@mrpevans

Put a Raspberry Pi-powered information display right on your desk using a stand you design and create yourself

For many, making is just plain fun. Often the enjoyment doesn't come from the finished product but the process of designing and creating it in the first place. That said, the world of making can seem intimidating to a novice. All these skills, terminology, and machinery! The fact is, anyone can be a maker and you don't (and indeed shouldn't) have to start big. The world of 3D printing has put fabrication in the hands of a whole new breed of maker – here we'll use it to design and build a simple stand for an e-ink weather display.

01 Plan your thing!

It seems an obvious place to start but when making, planning is everything. Just diving in to creating your awesome thing without any research or thoughtful design is a sure path to frustration. Don't worry about failure – that's fine, failure is how we learn. Just be careful to avoid wasting your own time and resources when a bit

of online searching, asking friends, or engaging with the community could save you hours. Your first questions should be 'has this been done already?' and 'does it really solve a problem I have?' (Although making something just *because* is absolutely fine.)

02 Get drawing

We designed a stand for a Raspberry Pi Zero W and Inky pHAT e-ink display so we could show the weather and time – a classic maker project. To keep things simple (always keep things simple if you can), our stand consisted of two parts: a mount for the electronics and a base to support it. We also added some standoffs so we could screw the assembly securely on to the mount. The first step was to sketch out some ideas and look around for inspiration. It's best not to rush in with your first idea – instead, try a few different approaches until you have something feasible.

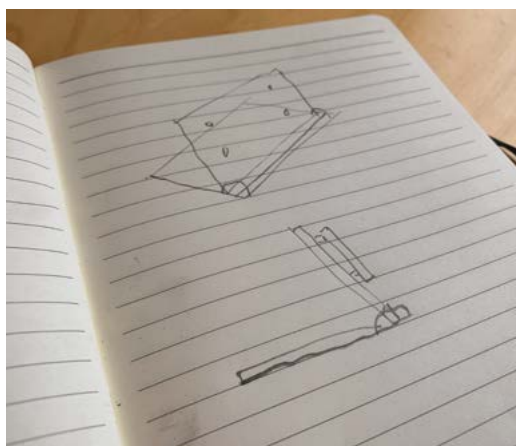
“ Try a few different approaches until you have something feasible ”

03 Measure twice, cut once

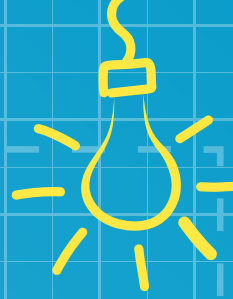
Time to grab the ruler and get some solid measurements. Measure up a Raspberry Pi Zero W and work out suitable dimensions from that. Room is needed for power; a micro-USB connector will need space at the bottom and ideally trail off behind the stand, so extra height is needed and a hole for the cable. Finally, the base needs a slot for the mount to sit in, so has to be angled and deep enough to hold everything securely (glue could always help out here).

You'll Need

- ▶ Inky pHAT display
magpi.cc/inkyphat
- ▶ 3D printer (or access to one), e.g.
magpi.cc/ender3
- ▶ PLA filament
3dprintz.co.uk
- ▶ 2 × 12 mm 2.5M bolts
- ▶ 4 × 2.5M nuts



▲ Start with your imagination and get some ideas down on paper. It doesn't matter how ridiculous; there are no rules at this stage



Top tip!

Here comes the remix

Many makers make their STL file available for remixing, so you can take an existing design and import it to Tinkercad for a head start.

▼ Now add physics. Measure everything up and make sure your design is feasible. Don't worry if it doesn't work out, just try again!

The mount and base are 3D-printed with PLA, a plant-based plastic that is strong and pliable but, most of all, not made from dinosaurs

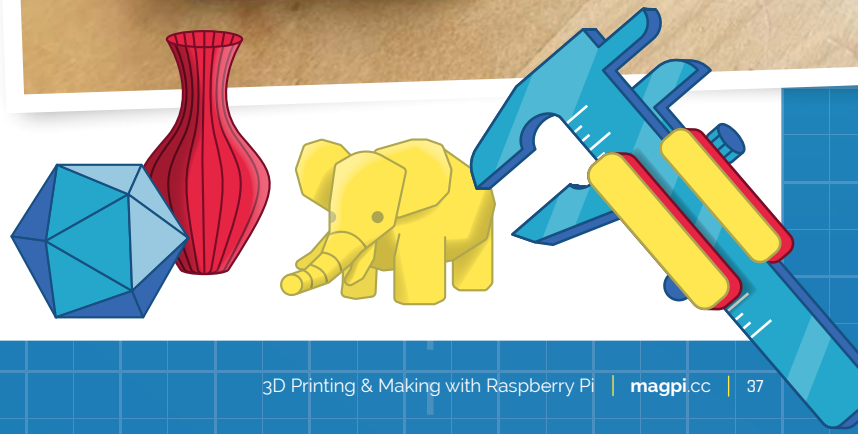
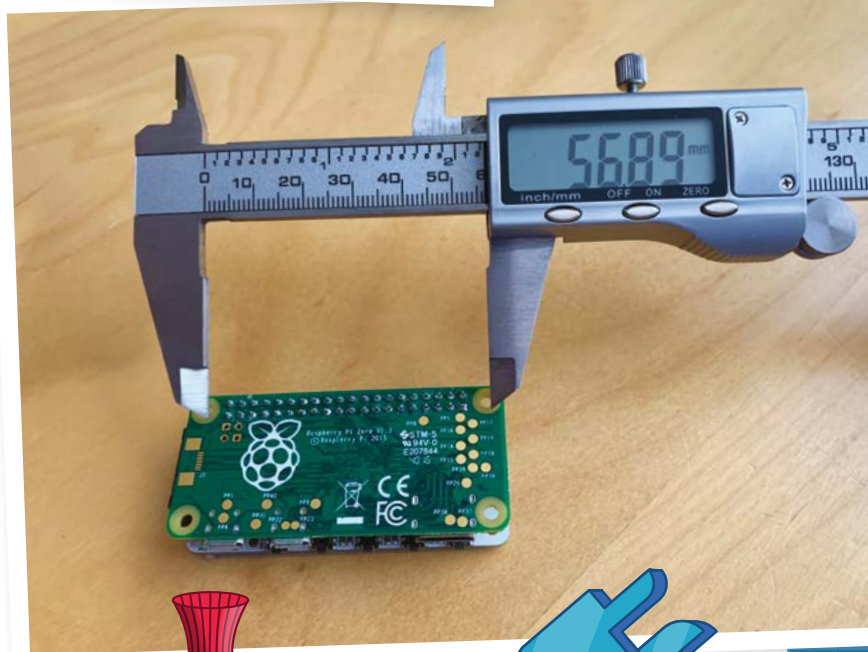
A slightly rotated cut into this wedge holds the mount in place, no glue required

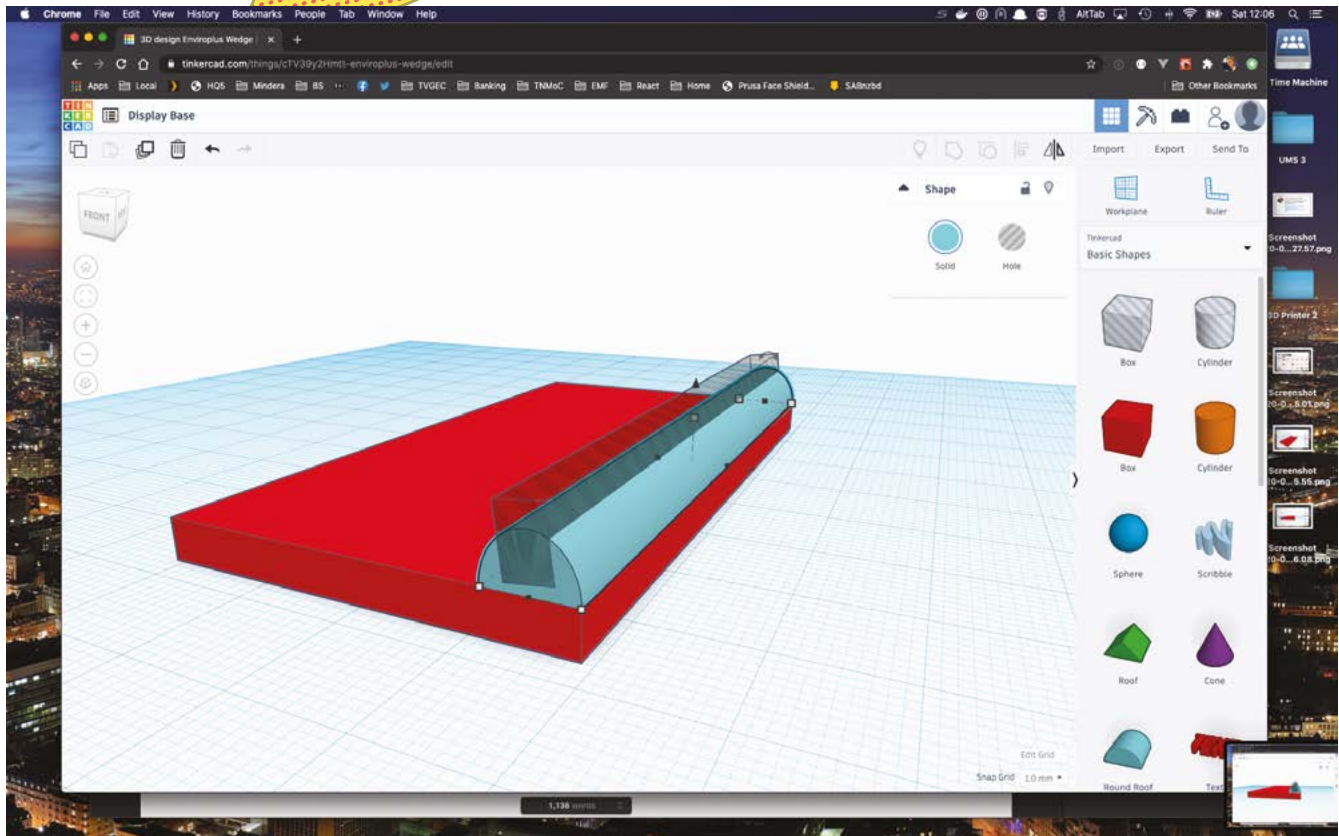
04 Analogue to digital

We can now model the design in a CAD (computer-aided design) package so we can create the files for 3D printing. Although professional packages like Fusion360 can have a steep learning curve, there are friendly offerings such as Tinkercad (tinkercad.com), a free online tool which is great for simple projects. It works by taking different geometric shapes and allowing you to piece them together to create a single object. You can also make holes by 'punching out' shapes from solid objects. With a little practice, you will soon be making complex shapes.

05 All about the base

To make the base, create a rectangle 40 mm × 75 mm × 4 mm and then add a 40 mm × 8 mm 'round roof' (kind of a semi-cylinder) along one long edge. Here's the clever bit: once the round roof is positioned in exactly the right place on the rectangle, selecting both and clicking 'group' turns them into a single object. The final step is to add the slot by creating a 10 mm × 75 mm × 4 mm rectangle, rotating it 7°, and then positioning it in the round roof. Finally, turn it into a 'hole' and group again. This has the effect of cutting out the shape from the round roof.





“ A slicer converts the STL into a series of instructions (G-code) that 3D printers can understand **”**

▲ Tinkercad is an amazing in-browser tool. Don't forget to check your measurements again and again so you don't lose time in printing

06 A panel that's hard to beat

Next, the mounting panel. This slots into the 'cut' made in the round roof section of the base. Starting with a rectangle 110 mm × 75 mm × 4 mm, carefully measure the screw-holes on your Raspberry Pi Zero W and create holes for the top two, each one 2.5 mm in diameter and 4 mm deep (so they go right through the mount). This is done by creating a cylinder, setting it to be a hole, and then placing it in the mount. It, and a copy, should be placed 10 mm from the top edge and exactly 57 mm apart (9 mm from each edge). This may take a few goes to get right. Finally, repeat this process, creating a larger hole (15 mm) that is 10 mm from the bottom and left edge. Select all and group to create the holes.

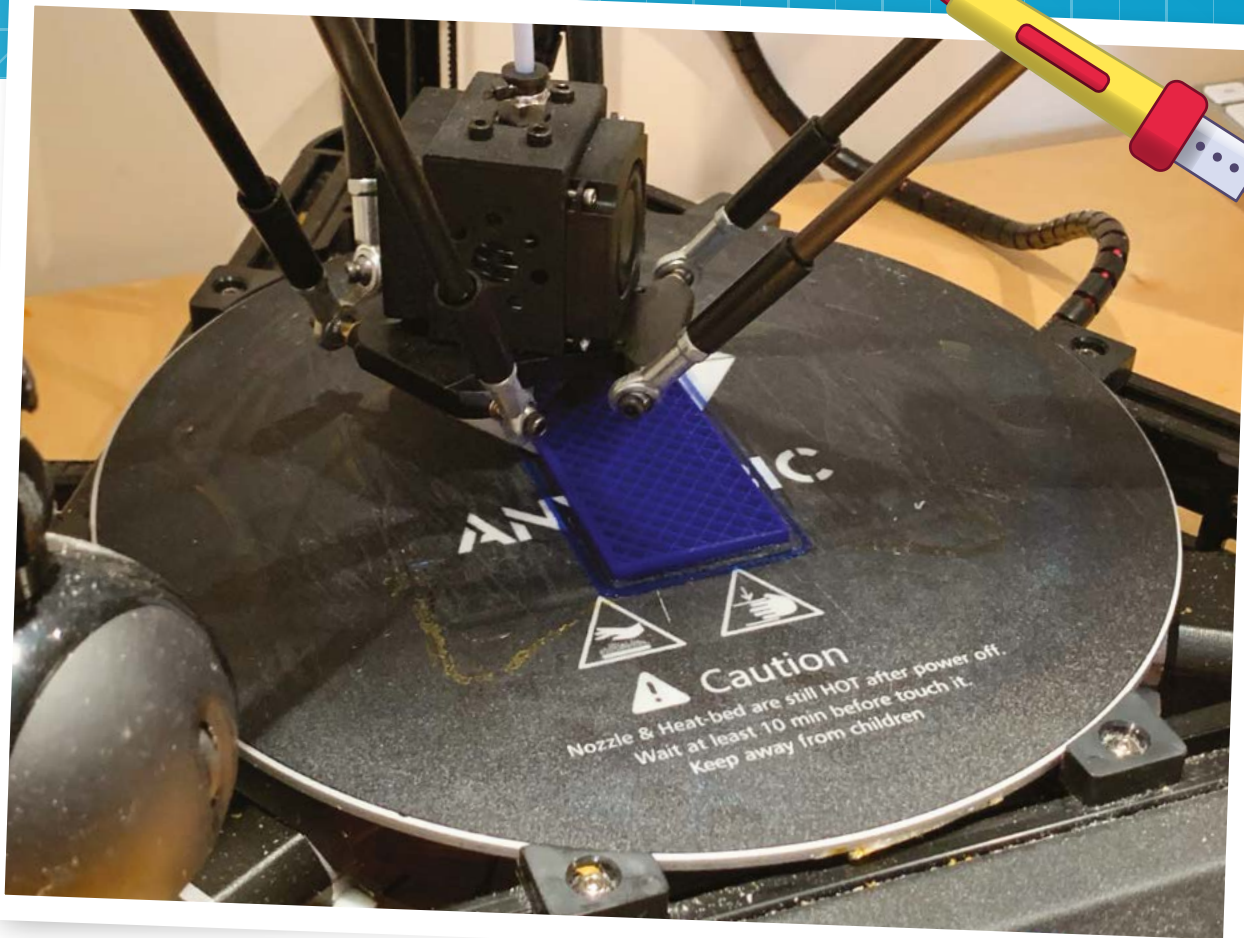
07 Fancy a slice?

By now you should have a rectangle with three holes in Tinkercad. Use the Export feature to download STL (Standard Tessellation Language) files of both the base and the mount. These are mathematical descriptions of the objects. To be able to 3D-print them, you need a slicer. This converts the STL into a series of instructions (G-code) that 3D printers can understand. The description is converted into printer operations that create the object layer-by-layer (hence 'slicing'). A popular (and free) choice is Cura. You can view your object and set your printing options. No supports or brims are required for these prints.

08 Time to print

Finally, the really fun part: 3D-printing your mount. This is two prints (although you have the option in Cura to print both in one session if you wish). Upload the G-code to your printer (how you do this will vary from printer to printer, but we're






a big fan of OctoPrint here, especially running on a Raspberry Pi). We printed our parts in PLA on an Anycubic Kossel Linear Pro at 0.2 mm resolution, which is a nice balance between quality and speed. Remember to check your levelling first and give the print time to 'settle' once complete. Finally, never leave a 3D printer unattended.

09 Finishing off

A disadvantage of the 3D printing method is the layering effect. Most 3D prints show a 'ripple' of the 0.2 mm layers printed. For many printed objects, this may not be a problem, but it is possible to create a smooth finish with some sandpaper and elbow grease. Starting with course sandpaper (100–200 grit), wear away the 'lines', increasing the grit used gradually up to about 600. For a 'perfect' finish, XTC-3D is a chemical coating designed specifically for 3D-printed objects that produces a silky-smooth finish. Just apply with a foam brush; normally, just one coat is all that's needed.

10 Assemble

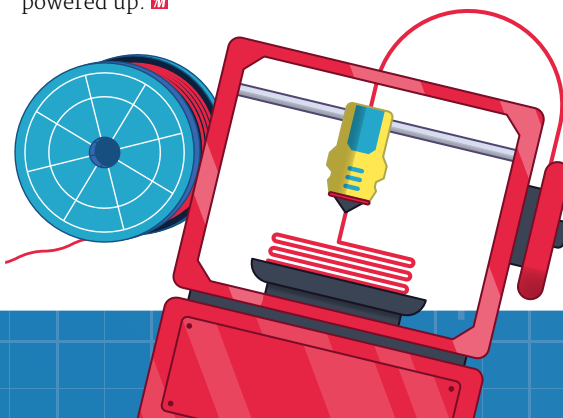
To create your finished product, make sure you've got your Raspberry Pi Zero W and Inky pHAT display all working. (Need some inspiration for the code? Look at magpi.cc/inkyphatdocs.) Secure the Raspberry Pi Zero W to the mounting panel using two 6 mm 2.5M bolts alongside the GPIO header. Use two nuts per bolt, one going each side of the mounting panel. Now mount the Inky pHAT to Raspberry Pi Zero W. Feed the USB power connector through the larger hole and plug in. Finally, insert the mounting panel into the base's slot. Your display is all ready to be powered up. 

▲ 3D printing does have a learning curve, but practice pays off and soon you'll be slicing and printing with confidence


Top tip!

Power up

If you're not finding the USB power pleasing to the eye, you could solder directly to the GPIO and create a smaller mount. Tricky, but not impossible.



DOSBox-X: Strictly legal game emulation



KG Orphanides

MAKER

KG enjoys making new computers pretend to be old ones, creating extremely niche games, and generating uncanny electronic screeching.

@KGOOrphanides

Use GOG, innoextract, DOSBox, and ScummVM to buy, extract, and run PC gaming classics on Raspberry Pi, from Zork to Blade Runner

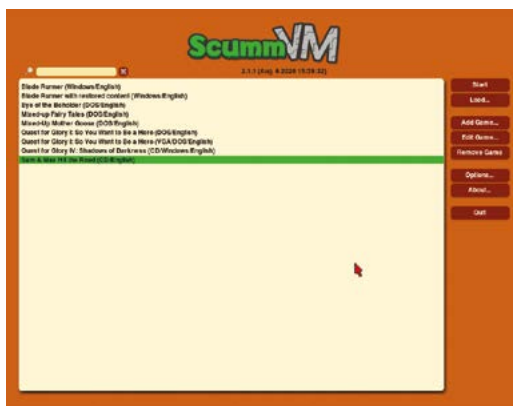
Games publishers have begun to recognise the value of their back catalogues. In some cases, that's spawned extravagant remakes, but in other cases, classic titles have just quietly reappeared to buy on digital storefronts.

It's those authentic classics that we're going to look this month: where to buy them and how to run them on Raspberry Pi. Although these titles mostly rely on the DOSBox and ScummVM emulators, which work perfectly well on Raspberry Pi OS, the games are packaged for installation on x86 hardware.

Fortunately, if you shop at GOG – formerly known as Good Old Games – Windows installers are created using the Inno Setup packager. A command-line tool called innoextract unpacks these so that you can just copy out the files of the game you've bought out, and point your emulator at them.

You'll Need

- DOSBox
dosbox.com
- DOSBox-X
(optional)
magpi.cc/dosboxx
- Innoextract
constexpr.org/innextract
- ScummVM
scummvm.org



▲ ScummVM originally just played LucasArts games, but the latest version can handle many more graphical and text adventures, as well as a few RPGs

01 Build innoextract

We're going to build innoextract from source as, at time of writing, the main Raspberry OS repo has an older version.

```
sudo apt install build-essential cmake
libboost-all-dev liblzma-dev
wget https://constexpr.org/innoextract/
files/innoextract-1.8.tar.gz
tar xzf innoextract-1.8.tar.gz
cd innoextract-1.8
mkdir -p build && cd build
cmake ..
make
sudo make install
```

“ The games are packaged for installation on x86 hardware ”

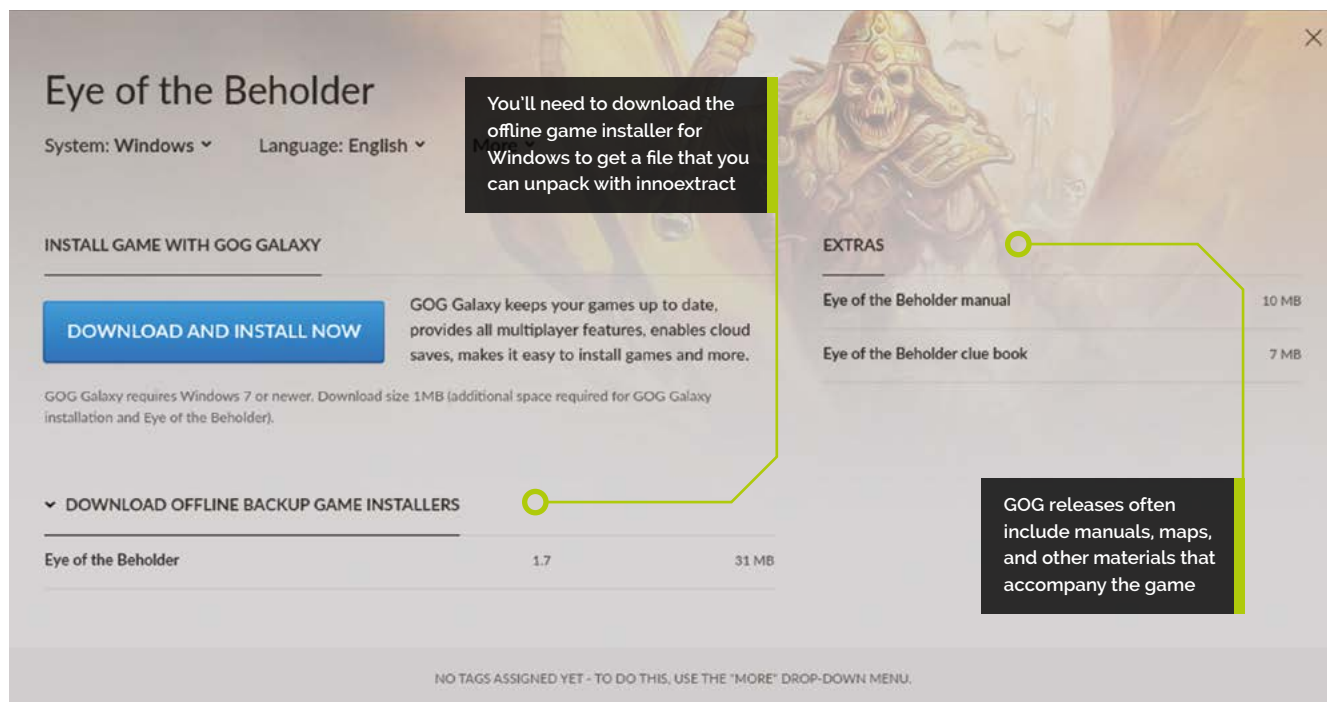
02 Version check

If your copy of innoextract is older than the file you're trying to extract, then the process may fail. At the Terminal, type:

```
innoextract -v
```

This will give you the version number and which Inno Setup files it can handle.

To confirm that you're using the current version, go to constexpr.org/innoextract to confirm what that is. If there's any disparity, download the latest source code and follow the instructions above.



03 Get a game

If you don't already have a GOG account, visit login.gog.com/register to create one. You'll have to be 16 or over. If you've got an account, log in and move on.

To make sure innoextract works, we're going to start by downloading one of the free retro games available on GOG. Go to magpi.cc/freegog to see a list of all the freebies on the service.

You'll probably want to grab all of these, but for now select Jill of the Jungle: The Complete Trilogy, click on the shopping cart icon towards the top right of the screen, and check out for zero pounds.

04 Download a game

Go to your games library at magpi.cc/gogaccount and click on – or, if you've got lots of games in there, search for – 'Jill'. On the download screen, make sure System is shown as Windows and ignore the prominent download button (that's for GOG's Galaxy manager).

Instead, click on the heading further down that says 'download offline backup game installers' to open up the file list. Now click on the line below it that says 'Jill of the Jungle: The Complete Trilogy'. While you're here, download the manual from the link towards the right of the download screen.



◀ Eye of the Beholder is a classic dungeon crawler, but remember to make a map to avoid getting lost in the sewers of Waterdeep

05 Extract a game

In a Terminal window, enter:

```
mkdir -p Games/{Jill,}
cd Downloads
cp setup_jill_of_the_jungle_1.0_cs_\
(28162\).exe /home/pi/Games/Jill/
cd /home/pi/Games/Jill
innoextract setup_jill_of_the_jungle_1.0_
cs_\(28162\).exe
```

The version number of this file could change, so it's easiest to just type `setup_jill` and press `TAB` to autocomplete the command when calling upon it.

06 Play a game

If you've not already installed DOSBox (or DOSBox-X) during one of our previous tutorials, you'll need it for this, so:

```
apt install dosbox
```

Once your DOS emulator is installed, type:

```
dosbox jill11.exe
```

Jill of the Jungle is a classic platformer from an era when shareware ruled the genre on MS-DOS. If you have a USB controller plugged in, press C on the initial setup screen to configure it – it's a bit easier to get used to than arrow keys if you're a regular WASD user.

The extraction process is the same for any of GOG's Inno Setup packaged games, which includes most of their DOS era titles. If you're working with games for Windows 95 through ME, such as Diablo, then

Games for free

Ensuring that your retro games are legally sourced can be a challenge online, but as well as digital storefronts, a number of free software sites are also committed to sharing only fully legit content, from shareware, homebrew, and demos to games made freeware by their publishers.

DOSGames.com

All vetted for licensing, you'll find everything from shareware FPS classics to game makers and freeware re-releases. A well-supplied kids' section and a violence warning system are handy for both parents and anyone who's all Grimdarked out.

dosgames.com

DOS Games Archive

While there's some replication between this and DOSGames, the inclusion of non-playable demos makes this archive an interesting stop for games history buffs.

dosgamesarchive.com

DOS Haven

People are still making games for DOS, and you can find almost 20 years' worth of those homebrew projects here, from the outré to the winningly sincere.

doshaven.eu

you may need to follow our Windows 98 emulation feature (magpi.cc/96). However, many now work smoothly with the ScummVM emulator, which runs natively on Raspberry Pi OS.

A hive of Scumm and villainy

If you want to play some games, rather than accurately emulating a full operating system, ScummVM, originally developed to play LucasArts point-and-click adventures such as The Secret of Monkey Island, is an outstanding tool with massive scope. Over the years, it's expanded to emulate virtually every graphical adventure game out there, and the latest versions even support classic RPGs and text adventures.

At the time of writing, ScummVM is at version 2.1.1, and we strongly recommend always using the latest version to ensure the widest possible support for newly added games. The current version available in Raspberry Pi OS's repositories is 2.0.0, which lacks some of the cooler features of its successor.

So let's head to the Terminal for a source install:

```
wget https://www.scummvm.org/frs/scummvm/2.1.1/scummvm-2.1.1.tar.gz
tar xzf scummvm-2.1.1.tar.gz
cd scummvm-2.1.1
apt install g++ make git libstdc++-dev liba52-dev libjpeg62-turbo-dev libmpeg2-4-dev libogg-dev libvorbis-dev libflac-dev libmad0-dev libpng-dev libtheora-dev libfaad-dev libfluidsynth-dev libfreetype6-dev zlib1g-dev libfribidi-dev libstdc++*
./configure
make clean
make -j4
make install
```



▲ You can set global options to be applied to every game – we recommend a graphical scaler – as well as individual game options via the Edit Game menu



▲ ScummVM was originally designed to run LucasArts/LucasFilm Games releases, such as Sam & Max Hit the Road

Even better for legal retronauts, you can find a range of classic games, re-released as freeware, to download at scummvm.org/games, including Revolution Software's Beneath a Steel Sky and Lure of the Temptress, as well as other once-commercial releases like Creative Reality's cyberpunk adventure DreamWeb and Interactive Binary Illusions' Flight of the Amazon Queen. You'll also find some of these in Raspberry Pi OS's official repositories. If you want them, enter:

```
sudo apt install lure-of-the-temptress
beneath-a-steel-sky dracula flight-of-the-
amazon-queen
```

Just as re-releases of retro DOS games are frequently bundled with a functional copy of DOSBox, point-and-click re-releases often use ScummVM. When extracting files bought from GOG, you can delete or ignore the Windows version of the emulator that's included and just point your own at the game files.

“GOG isn't the only place to buy retro games, but it has a wide range of titles”

Top tips

Remember to check that the game you want to play is supported by your emulator of choice. DOSBox and ScummVM publish lists of supported titles, and GOG store pages list the emulator they use at the very bottom.

Make sure the re-release includes any manuals, maps, and multimedia material you'll need – some digital re-releases have been created with more care than others, and manual-based copy protection was common in the DOS era.

GOG isn't the only place to buy retro games, but it has a wide range of titles and its installers are

Three to buy

Loom | Price: £5

With a soundtrack and imagery inspired by Tchaikovsky's *Swan Lake*, Brian Moriarty's Loom is a beautifully woven adventure game with a compelling magical setting. The CD version has full speech, but there's a bit more game to play if you opt for the EGA original.

magpi.cc/gogloom

Eye of the Beholder | Price: £8

A classic of the flick-screen dungeon crawling genre, Eye of the Beholder is one of the most approachable Advanced Dungeons & Dragons RPGs and oozes atmosphere. Perhaps unexpectedly, it works perfectly through ScummVM, making for a particularly easy emulation experience. Both its sequels are included in the bundle.

magpi.cc/gogeoth

Blade Runner | Price: £8


As dark and gritty as *Blade Runner*'s cinematic universe, and features some of the cast from Ridley Scott's 1982 film, the Blade Runner adventure game was for years effectively lost to licensing disputes and lack of format support until it appeared in GOG's catalogue in 2019. It's a unique and much-lauded work that contributes to the series' mythos in its own right, but carries content warnings off the bat for extremely mature subject matter.

magpi.cc/gogrunner

easy to extract. However, if you already own some classics on Steam, there's generally no DRM on these, so you can simply copy their files over to Raspberry Pi.

Use the biggest microSD card you can. As games move into the full-speech era, you'll find plenty that come in at around a gigabyte each.

'Abandonware' is a term used online for old software that players believe has been 'abandoned' by its rights holders. Although this scene has been important for games preservation, there's no legal basis for abandonware in most countries, and downloading it is still software piracy. We stick to legal software.

'Homebrew', on the other hand, is new, independent, and generally non-commercial software made for traditionally closed or nominally defunct platforms. While homebrew titles can often be played on original hardware, they're ideal for emulation. They're generally fully legal, although some fan games have been known to appropriate graphical assets. 

High Quality Camera: Flash photography using an LED

Add an LED flash to shoot images in low light

You'll Need

- Camera Module / HQ Camera
- White LED
- Resistor

The Raspberry Pi Camera Module or HQ Camera works really well in good lighting conditions, but what if there's less light available? Here, we show you how to set up a simple LED flash, which will be triggered each time you take a photo, using the picamera Python library. We also take a look at how to shoot better images in low light when you are not using a flash.

01 Download device tree source

Before we can wire up a flash, we need to configure a GPIO pin to use for it. This will then be triggered each time we capture a still using picamera with the flash mode set to on. To do this, we need to edit the VideoCore GPU default device tree source. First, install device tree compiler with:

```
sudo apt-get install device-tree-compiler
```

Then grab a copy of the default device tree source with:

```
wget https://raw.githubusercontent.com/raspberrypi/firmware/master/extra/dt-blob.dts
```

02 Edit the device tree source

Edit the file using your favourite text editor, such as nano:

```
sudo nano dt-blob.dts
```

You'll need to find the correct part of the code for the Raspberry Pi model you're using; for instance, the part for Raspberry Pi 4 is found under `pins_4b {`.

Here you'll find `pin_config` and `pin_defines` sections. In the `pin_config` section, add a line to configure the GPIO pin (we're using GPIO17) that you want to use for the flash:

```
pin@p17 { function = "output"; termination = "pull_down"; };
```

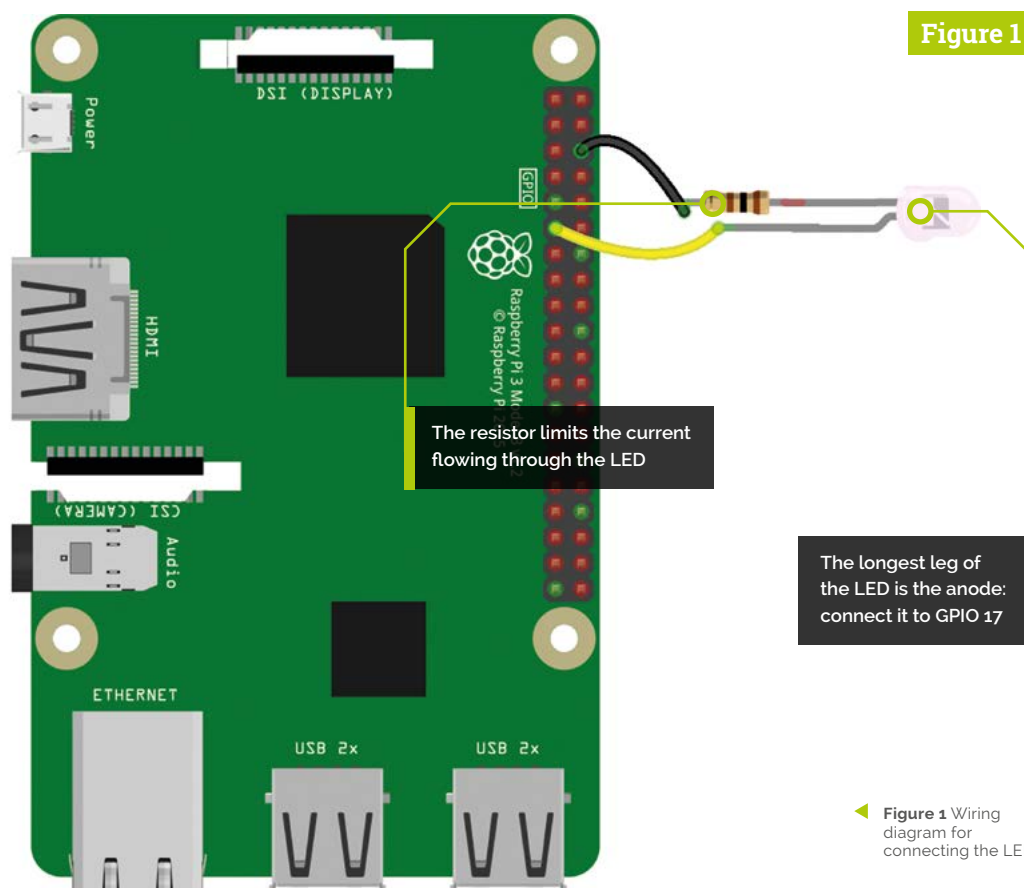
“ Before we can wire up a flash, we need to configure a GPIO pin to use for it ”

03 Enable flash

Next, we need to associate the pin we added with the flash enable function by editing it in the `pin_define` section. We simply change `absent` to `internal` and add a line with the pin number, so it looks like the following:

```
pin-define@FLASH_0_ENABLE {
    type = "internal";
    number = <17>;
};
```

Note that it's the `FLASH_0` section that you need to alter: `FLASH_1` is for an optional privacy LED to come on after taking a picture, but we won't bother with that.



04 Compile the blob

With the device tree source updated, we now need to compile it into a binary blob, using the following command in a Terminal window:

```
dtc -q -I dts -O dtb dt-blob.dts -o dt-blob.bin
```

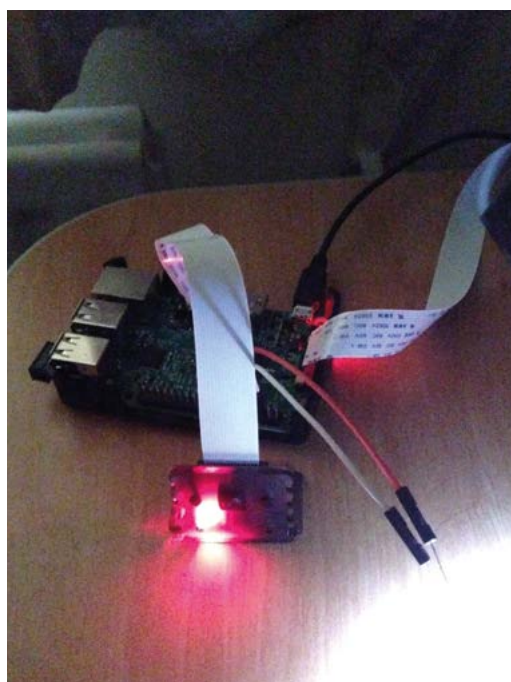
This should output nothing. Next, you need to place the new binary on the first partition of the microSD card. In the case of non-NOOBS Raspbian installs, this is generally **/boot**, so use:

```
sudo cp dt-blob.bin /boot/
```

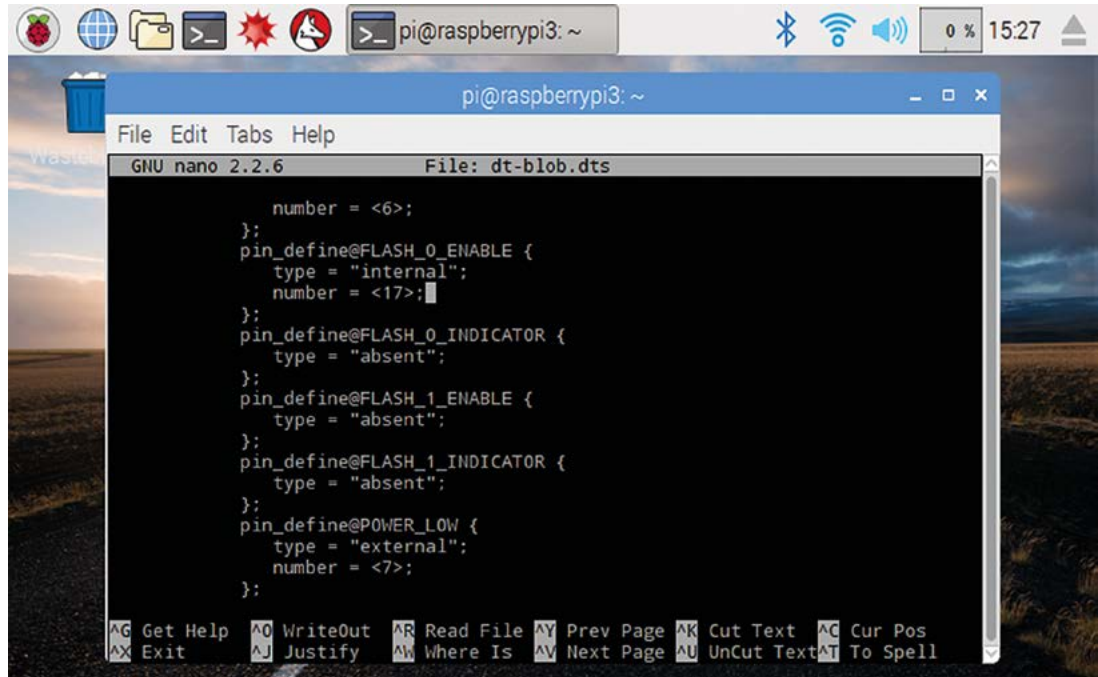
In you installed Raspbian via NOOBS, however, you'll need to do the following instead:

```
sudo mkdir /mnt/recovery
sudo mount /dev/mmcblk0p1 /mnt/recovery
sudo cp dt-blob.bin /mnt/recovery
sudo umount /mnt/recovery
sudo rmdir /mnt/recovery
```

To activate the new device tree configuration, reboot your Raspberry Pi.



▲ Wire the white LED to GPIO 17 and GND via a low-ohmage resistor



► You need to edit the device tree source to enable a GPIO pin for the flash

05 Wire up the LED

Connect a white LED – we used a 5 mm one – to your Raspberry Pi as in **Figure 1** (previous page). The LED’s anode (long leg) is connected to our flash-enabled pin, GPIO17. To be sure of the LED not burning out from excess current, you should add a low-ohmage resistor (such as 100Ω) between the LED’s cathode (short leg) and Raspberry Pi’s GND pin. Depending on the maximum forward voltage of your LED (ours was 3.5 V), you may want to choose a resistor with a different ohmage, but it’s always best to use one to be safe.

If you want to use higher-powered or multiple LEDs, you’ll have to think about powering them via a suitable driver circuit, with a transistor wired to the flash pin. You may also need a separate power supply. Note that, due to the Raspberry Pi Camera’s rolling shutter, only an LED or equivalent flash is suitable: you can’t use a xenon flash. Alternative flash/lighting methods include NeoPixel sticks and the LISIPAROI light ring.

06 Test it out

With the LED connected, we can now test out our flash with a short Python program. In Thonny, create a new file and enter the code from **listing1.py**. The `camera.flash_mode = 'on'` line

sets the flash to trigger when we issue the capture command below; the LED will light up briefly before the image capture, to enable the camera to set the correct exposure level for the extra illumination, before the flash proper is triggered. If you want the flash to trigger automatically only when it’s dark enough, you can change the penultimate line of the code to `camera.flash_mode = 'auto'`.

“ The LED will light up briefly before image capture, so the camera can set the correct exposure level ”

07 Low-light photography

In low-light scenarios where you don’t want to use a flash, you can improve capture of images using a few tricks. By setting a high gain combined with a long exposure time, the camera is able to gather the maximum amount of light. Note that since the `shutter_speed` attribute is constrained by the camera’s frame rate, we need to set a very slow frame rate. The code in **listing2.py** captures an image with a six-second exposure time: this is the maximum time for the Camera Module V1 – if you



▲ Even a single white LED can provide illumination for close-up photography

have a v2 Camera Module, it can be extended to ten seconds, or much longer for an HQ Camera. The frame rate is set to a sixth of a second, while we set the ISO to 800 for greater exposure. A pause of 30 seconds gives the camera enough time to set gains and measure AWB (auto white balance).

Try running the script in a very dark setting: it may take some time to run, including the 30-second pause and about 20 seconds for the capture itself. Note: if you're getting a timeout error, you may need to do a full Raspbian upgrade with `sudo apt update` and `sudo apt dist-upgrade`.

The particular camera settings in this script are only useful for very low light conditions: in a less dark environment, the image produced will be heavily overexposed, so you may need to increase the frame rate and lower the shutter speed accordingly.

If the image has a green cast, you'll need to alter the white balance manually. Turn AWB off with `camera.awb_mode = 'off'`. Then set the red/blue gains manually; e.g. `camera.awb_gains = (1.5, 1.5)`. [\[M\]](#)



▲ Using a long exposure, you can shoot stills in very dark settings

listing1.py

► Language: **Python 3**

**DOWNLOAD
THE FULL CODE:**



magpi.cc/github

```
001. import picamera
002.
003. with picamera.PiCamera() as camera:
004.     camera.flash_mode = 'on'
005.     camera.capture('foo.jpg')
```

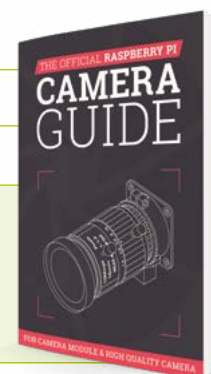
listing2.py

► Language: **Python 3**

```
001. from picamera import PiCamera
002. from time import sleep
003. from fractions import Fraction
004. # Set a framerate of 1/6fps, then set shutter
005. # speed to 6s and ISO to 800
006. camera = PiCamera(resolution=(1280, 720),
007.     framerate=Fraction(1, 6))
008. camera.shutter_speed = 6000000
009. camera.iso = 800
010. # Give the camera a good long time to set gains and
011. # measure AWB (you may wish to use fixed AWB
012. # instead)
013. sleep(30)
014. camera.exposure_mode = 'off'
015. # Finally, capture an image with a 6s exposure. Due
016. # to mode switching on the still port, this will take
017. # longer than six seconds
018. camera.capture('dark.jpg')
```

The Official Raspberry Pi Camera Guide

For further tutorials on how to use your HQ Camera or Camera Module, take a look at our new book, *The Official Raspberry Pi Camera Guide*. Its 132 pages are packed with essential info and a range of exciting projects.
magpi.cc/cameraguide



Build the ultimate home music system



PJ Evans

PJ is a writer, software engineer and can't decide whether to arrange his collection by genre, title, or year of release.

@mrpjevens



**Warning!
Copyright**

Where you live may or may not allow the copying of data from a CD. Check here for more info:

magpi.cc/ripping

You'll Need

- Mopidy Music Server magpi.cc/96

Extend the capabilities of your music system with Mopidy extensions and access a world of audio

Last month we built a standalone music server using a DAC (digital-to-analogue converter) HAT, Mopidy, Iris, and a Raspberry Pi touchscreen. All these things, together with a pair of decent speakers, guarantee audio bliss for your home or workplace. Now we're going to extend the capabilities of Mopidy using its plug-in architecture. With a few commands, Mopidy can be extended to stream audio from a wide range of online services, from giants like Spotify to the gargantuan collections of the Internet Archive. This month we're looking at the best Mopidy extensions and how to install them.

01 Local library

Let's start by looking again at our local collection. Storing music locally allows you reliable playback and complete control over how your music is organised. The downside is storage costs, so before embarking on ripping all your CDs, have a think about how you want to store them. Last month we discussed using abcde (A Better CD Encoder) to rip to FLAC, which produces large files with no quality loss. You can also consider using Ogg Vorbis. This open-source codec does involve quality loss, but produces much smaller files, and there's every chance you won't be able to tell the difference. To rip a CD to Ogg Vorbis:

```
abcde -o ogg
```

02 Tag, you're it

Many CD rippers, abcde included, will use online services to find information about your CD and automatically name files after the artist,



▲ Listen to all your favourite podcasts directly from Mopidy and discover new shows with the iTunes Podcast catalogue

album, and track name. They will also record that information inside the file itself so servers like Mopidy can display details about what's playing. This is called *tagging*. Unfortunately, the online services don't always get it right, so if you find a tag is misnamed or contains mistakes, you can use Kid3 to set the tags yourself. To install from the command line:

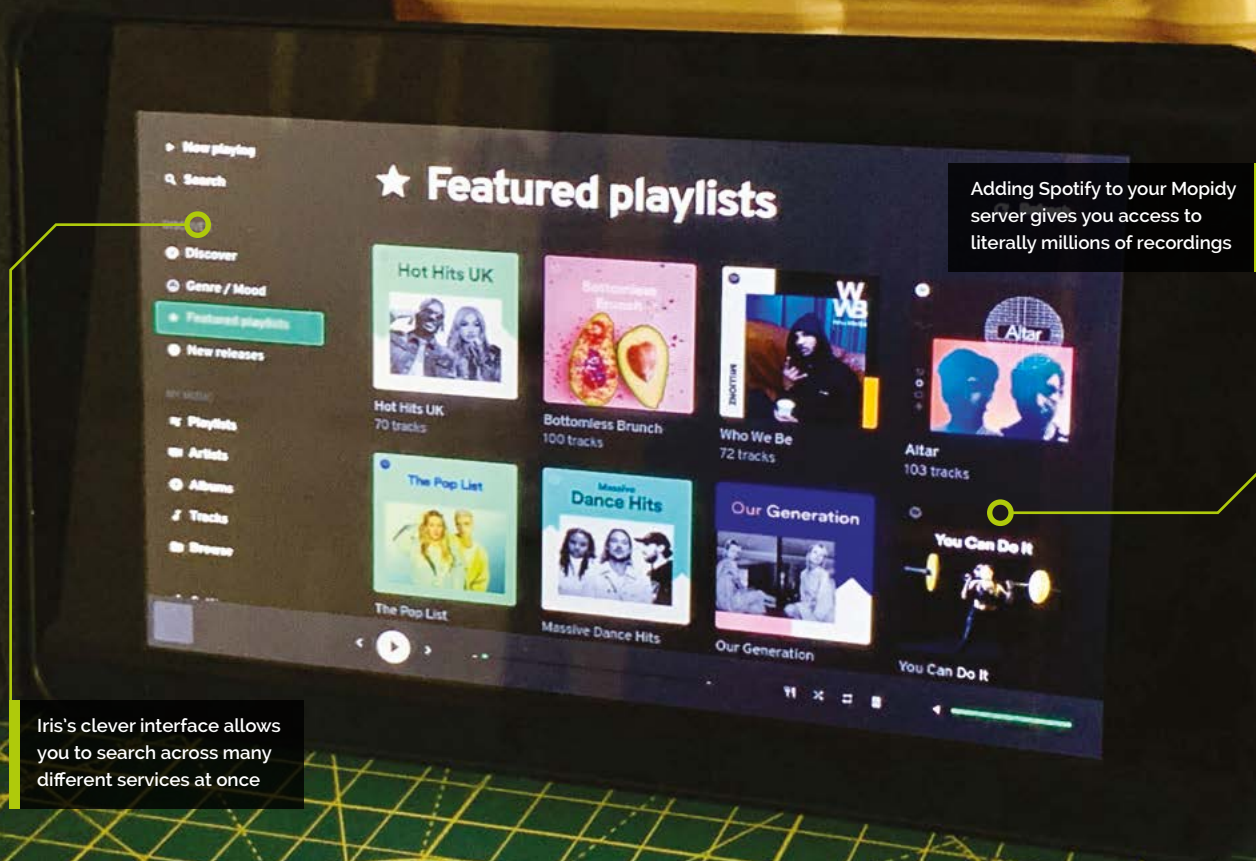
```
sudo apt install kid3
```

This desktop app provides easy bulk retagging of files and can even import artwork.

“ If you find a tag is misnamed, you can use Kid3 to set tags yourself ”

03 Figuring configuring

In the following steps, we'll be setting up a number of online services to interface directly with your Mopidy installation. In many cases, you'll need to make changes to the Mopidy config file,



so it's worth explaining how it's formatted. Each extension is named 'Mopidy-Extension'. In the config file located at `/etc/mopidy/mopidy.conf`, *Extension* becomes `[extension]` and any relevant options go underneath that line in the form 'key = value'. For example:

```
[local]
media_dir = /home/pi/Music
```

We've provided an example config file to help make sure things are properly set up.

04 Spot on

Now your local collection is set up, let's move on to the first of our extensions. Spotify is an amazing collection of music. Wouldn't it be great to be able to access not only Spotify's archive but also your playlists and tagged artists? If you've got a Spotify Premium account, you can do just that. Start by installing the extension from the command line:

```
sudo apt install mopidy-spotify
```

We now need to authorise Mopidy to use your account by visiting magpi.cc/mpdyspotify. Once you're authorised, a config file will be displayed.

Cut and paste this into `/etc/mopidy/mopidy.conf` and add your Spotify username and password. See the code listing overleaf for an example.

Now restart Mopidy:

```
sudo systemctl restart mopidy
```

Go back to Iris and the world of Spotify is literally at your fingertips.

05 Radio waves

How about adding radio stations to your Mopidy setup? There is easy and free access to thousands of radio stations streaming across the planet, catering for all interests. We have the option of two extensions for playing radio: TuneIn and RadioNet. The TuneIn extension provides access not only to live stations, but also historical recordings and is more suited to the curious browser rather than just getting to your favourite station. RadioNet is a simpler but easier-to-navigate service that allows you to create a list of favourite stations. To install:

```
sudo apt install mopidy-tunein
sudo pip3 install Mopidy-RadioNet
sudo systemctl restart mopidy
```

Top Tip

Lossless or lossy?

Lossless (FLAC) compresses audio with no loss of quality. Lossy (Ogg Vorbis, MP3) removes audio information to produce smaller file sizes.

Top Tip

Enabled or disabled?

Every extension supports 'enabled' in its config. If you want to remove something temporarily, just add 'enabled = false' to the extension's block in `/etc/mopidy/mopidy.conf`.

06 Casting out

If you prefer your radio programmes 'on demand', you're probably a fan of podcasts. You've got two options for listening to your favourite shows on Mopidy. If you like to subscribe direct to podcast RSS feeds, then it's as easy as this:

```
sudo pip3 install Mopidy-Podcast
sudo systemctl restart mopidy
```

To use it, you need to point to an OPML file (a list of podcast locations), as shown in the listing. Most podcast clients can generate this for you. If you prefer to browse a huge catalogue of programming, then additionally install the iTunes plug-in:

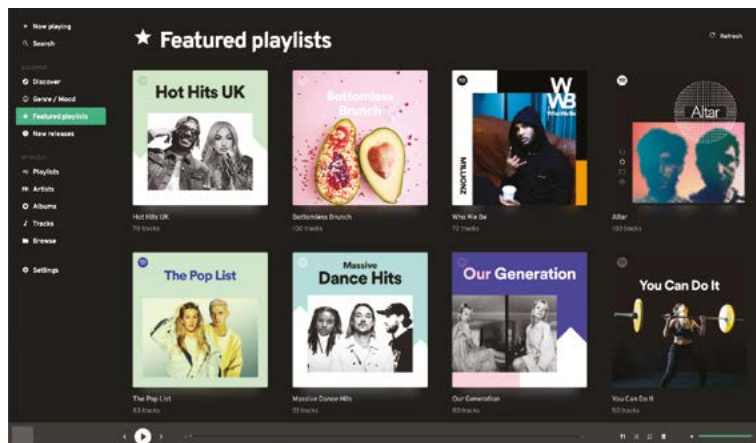
```
sudo pip3 install Mopidy-Podcast-iTunes
sudo systemctl restart mopidy
```

Now you can use Browse to find your favourite show and listen on demand.

07 SomaFM

SomaFM was one of the early pioneers of internet audio streaming. Working through a donations model, it has expanded to over 30 channels, all with the theme of ambience but covering different genres from electronica to jazz. Often SomaFM makes for perfect background music. To add SomaFM to Mopidy (you've probably spotted a pattern emerging by now):

```
sudo pip3 install Mopidy-SomaFM
sudo systemctl restart mopidy
```



▲ Add audio services, like Spotify, SomaFM, and YouTube for a fully-fledged modern music system

08 SoundCloud

SoundCloud is an excellent online showcase of music and other forms of audio. As well as providing a way for bands to sell direct to their fans, it's a platform for new and upcoming material from unsigned musicians. You can browse SoundCloud's vast collection from Mopidy, but as with Spotify, you need to have an account (it's free to join) and authorise your installation.

First, visit mopidy.com/ext/soundcloud and follow the instructions to authorise Mopidy. Copy and paste the resulting config file, and add it to `/etc/mopidy/mopidy.conf`. Now install the SoundCloud extension:

```
sudo pip3 install Mopidy-SoundCloud
sudo systemctl restart mopidy
```

“ You can browse SoundCloud's vast collection from Mopidy ”

09 YouTube

The ubiquitous YouTube contains countless hours of audio-specific content. Either listen to one of the many audio recordings uploaded or just the soundtrack of a favourite video. Again, the standard installation pattern applies:

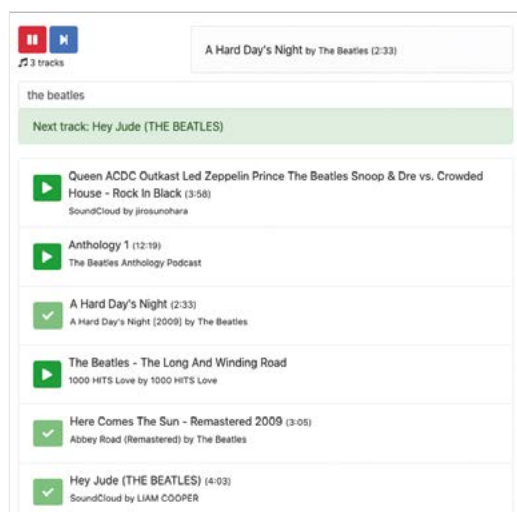
```
sudo pip3 install Mopidy-YouTube
sudo systemctl restart mopidy
```

One difference with this extension is the need to specifically enable it in `/etc/mopidy/mopidy.conf`. Make sure this is in the file or YouTube files will not appear in search results:

```
[youtube]
enabled = true
```

10 Digging through the archives

The Internet Archive (archive.org) is one of the crowning achievements of the internet. This vast resource aims to preserve anything that can be expressed in digital form. Unsurprisingly, its audio collection is vast, containing thousands of hours or public domain recordings, from vintage music to public speeches, documentaries, and a massive



▲ Mopidy-Party lets your friends choose the music, which may or may not be a good thing

catalogue of Grateful Dead recordings! You can access all of these from your Mopidy system:

```
sudo pip3 install Mopidy-InternetArchive
sudo systemctl restart mopidy
```

We recommend using the 'Browse' feature of Iris to navigate this resource. Search results can be overwhelming!

11 Party time, excellent!

You now have access to a mind-boggling array of music and audio. Time to celebrate. We've been using Iris as our interface to Mopidy, but in fact you can install different interfaces and even run them concurrently. If you're having a few friends around, Mopidy-Party allows you to safely give everyone access to your music collection from their phones and add their picks to a playback queue. Now, who put Showaddywaddy on?

12 Make your own

Something missing from your collection? Is there no access to the latest and greatest streaming service? If you're handy with a bit of Python coding, then you're free to make your own Mopidy extensions and share them with users all over the world. Not only have the Mopidy developers provided detailed documentation at magpi.cc/mpdyext, they have also provided a 'cookiecutter' project at magpi.cc/mpdycookie which will get you up and running in no time. Contributions to improve existing extensions are also welcomed. [\[M\]](#)

mopidy.conf

► Language: **Config file**

**DOWNLOAD
THE FULL CODE:**



magpi.cc/modipyconf

```
001. [http]
002. # Make sure the web interface can be accessed by the
    local network
003. hostname = 0.0.0.0
004.
005. [audio]
006. # Direct audio to the DAC, not HDMI
007. output = alsasink
008.
009. # The File extension isn't of much use, so we can disable it
010. [file]
011. enabled = false
012.
013. # If enabled, this adds support for M3U playlists
014. [m3u]
015. enabled = false
016.
017. # Location of local music
018. [local]
019. media_dir = /home/pi/Music
020.
021. # Replace with the information given when you authorise
022. # then add in you Spotify username and password
023. [spotify]
024. username = <Spotify username>
025. password = <Spotify password>
026. client_id = <Client ID>
027. client_secret = <Client secret>
028.
029. # Getting to many results in TuneIn? Add this so you only
    get stations.
030. [tunein]
031. filter = station
032.
033. # Once authorised, add the provided token here
034. [soundcloud]
035. auth_token = <Auth token>
036.
037. # Most extensions are enabled by default. You must
    explicitly enable YouTube.
038. [youtube]
039. enabled = true
```

Home monitoring dashboard



Wesley Archer

Long-time Raspberry Pi fan and maker. From retro gaming to home automation, Wes loves a new project, especially when a Raspberry Pi is involved!

@raspberrycoulls

It's well known that a variety of sensors work with a Raspberry Pi, but why not monitor the data in a simple dashboard?

One of the best things about Raspberry Pi is the range of accessories and sensors, making it the perfect choice for real-world data monitoring projects. In this tutorial, we'll look at a couple of affordable and readily available sensors that easily connect to Raspberry Pi via the GPIO pins – the BME280 for temperature, humidity and pressure, and the TSL2561 for light levels. We'll then log the data via an online dashboard because everybody loves a graph! Who knows, over time maybe you can identify trends, or use the data to your advantage as part of a smart home setup?

01 Prototype your circuit first!

It is always a good idea to prototype your circuits using a breadboard and jumper cables first. This means you can ensure that everything works as expected before you start soldering and making more permanent connections. As the BME280 and TSL2561 sensors use the I²C pins on



You'll Need

- ▶ BME280 breakout sensor
- ▶ TSL2561 breakout sensor
- ▶ Jumper wires and breadboard
- ▶ Beebotte account beebotte.com

- ▶ We love the BME280 sensor because it is small, cheap, and accurate. It measures temperature, humidity and pressure, and only needs four wires too!

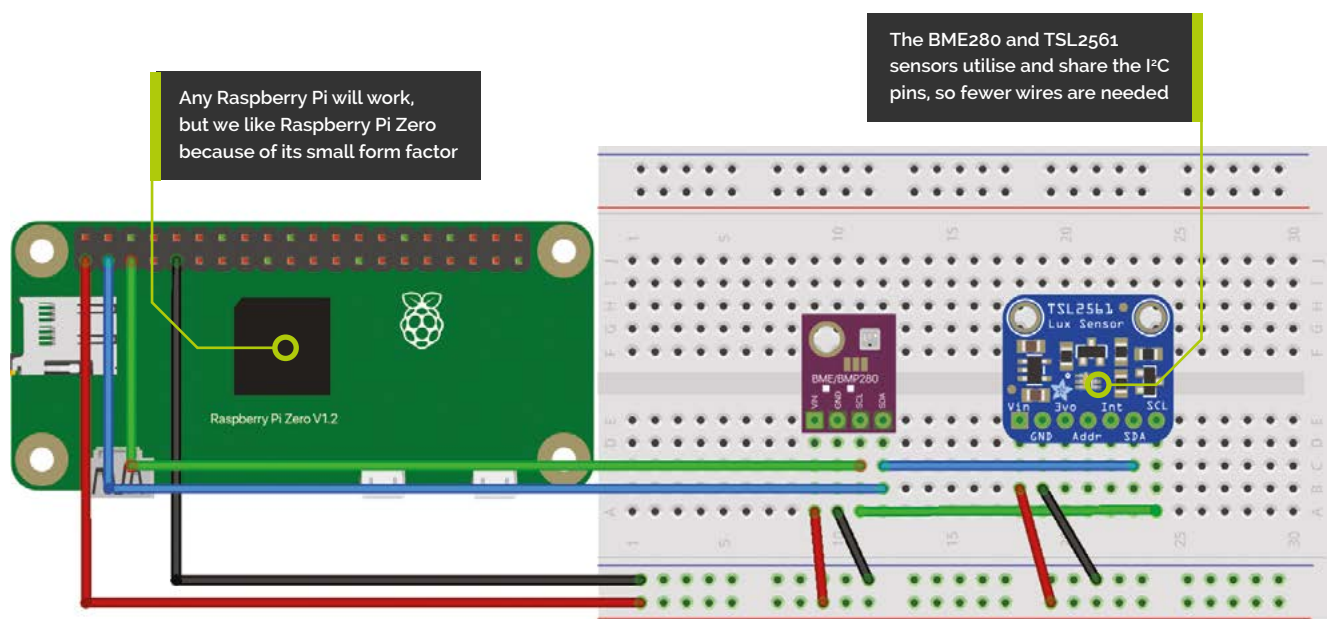
your Raspberry Pi, connecting them is very simple using the SDA and SCL pins on the GPIO (BCM 2 and BCM 3 respectively). Because I²C is used, the pins (and wires) can be shared, so the two sensors connect to the same pins on Raspberry Pi itself, meaning less soldering and wiring!

02 Create your Beebotte account

Beebotte is a cloud Platform-as-a-Service (PaaS) designed for IoT projects – best of all, it has a free account tier which provides more than enough service for what we need. Signing up is easy and can be done via beebotte.com and entering your details. With the free plan, we have to be aware of API limits, but the code in our tutorial has been optimised so that the usage is not exceeded at any time. However, if you want to place multiple Raspberry Pi computers with accompanying sensors in various rooms in your home, you may want to consider upgrading to a paid plan for the extra headroom.

03 Create a new channel in Beebotte

The first step to creating a dashboard in Beebotte is creating a channel so that your data can be stored when it has been sent by your Raspberry Pi. Once you have created your Beebotte account, you should see the 'My Channels' dashboard. If not, click the 'Channels' menu option on the left. Next, click 'Create New' and then you'll need to provide some details. The first one is the channel name, so you know what it's for, then you need to add your resources – i.e. temperature, humidity, pressure, and luminosity – which will be the data sent from our sensors.



04 Copy your Beebotte channel token

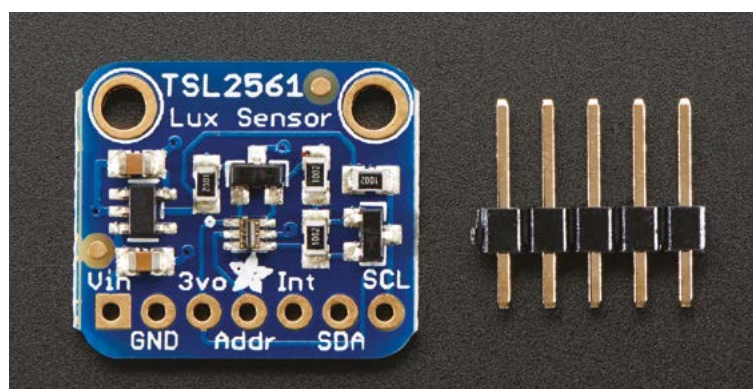
In order to send data to our channel via our Raspberry Pi, we'll be using the Beebotte API and in order to access your channel, you'll need to copy your unique channel token. This is created when you create your channel in the previous step, but you'll need to take note of this as you'll need to add it to our code so that you can communicate with your own channel directly. *You should never share this with anybody for security reasons*, but the token usually looks similar to `token_abcdefghijklmnop` and is shown when you view the channel after creating it.

05 Download our code

Now that you have a Beebotte account and a channel for the data to be sent to, you'll now need to download our code to your Raspberry Pi. This is available in GitHub, and you can clone the repository by typing:

```
git clone https://github.com/raspberrycoulis/bme280-tsl2561-beebotte.git
```

...after accessing your Raspberry Pi via SSH. If you haven't done this before, you may need to install Git on your Pi, which is as easy as entering `sudo apt-get install git`. This shouldn't take too long,



▲ The TSL2561 measures luminosity – the amount of light present – and also uses the I²C pins, so can share wires and pins with the BME280 sensor

but should leave you with all the code needed for the next steps.

06 Add your channel token and name

Right now, you should have the Beebotte account, channel, and the bare-bones code. For it to work, however, you'll need to make a few minor changes to the code so that your Raspberry Pi can communicate with your Beebotte account and channel. Remember that token in Step 4? Well, you'll need this now! Assuming you are in the directory where the code is stored, enter `nano dashboard.py` to open the Nano editor. Then you'll

Top Tip

Running our code automatically

Check out the read-me file in our GitHub to see how to run the code automatically on boot.

Create a new channel

MagPi

A channel to receive data sent by BME280 and TSL2561 sensors for temperature, humidity, pressure and luminosity respectively.

☒ Public

Configured Resources			
temperature	Temperature in celsius	number	<input type="checkbox"/> SoS
humidity	Relative humidity in percent	number	<input type="checkbox"/> SoS
pressure	Atmospheric pressure in hPa	number	<input type="checkbox"/> SoS
luminosity	Luminosity in lux	number	<input type="checkbox"/> SoS

[+ Resource](#)

Cancel [Create channel](#)

Resource Data Type

When creating a new resource, you need to specify the type of data that is expected to be persisted to this resource. Beebotte has a number of defined data types; click [here](#) for more information.

Send on Subscribe (SoS)

A new option called **Send on Subscribe (SoS)** has been added to the channel resource model. When enabled, this option allows the most recent persistent value to be automatically sent over when a client subscribes to the resource. By default, this option is disabled.

▲ A channel in Beebotte is required to receive the data measured by our sensors and sent by our Raspberry Pi so it can be visualised in a graph

need to find the parts that say **CHANNEL_TOKEN** and **YOUR_CHANNEL_NAME** and replace them with your token and channel name. Once done, hit **CTRL+X** then **Y** to save and exit.

firstly, it reads data from both the BME280 and the TSL2561 sensors. Then, the data is rounded to the nearest whole number and then it is sent to your Beebotte channel to be recorded. Finally, the code will go to sleep for 15 minutes – to avoid API limits – before repeating until stopped manually.

Top Tip

Always test circuits first

Breadboards are fantastic for testing your circuits before getting your soldering iron out. It's better to be safe than sorry!

07 Test it out

Assuming your channel token and channel name has been entered correctly – be sure to keep the quotation marks in the code – you should now be ready to test that your Raspberry Pi and your Beebotte account can communicate with each other. With the channel page open in your web browser, type **python dashboard.py** in your Raspberry Pi Terminal. After a few seconds, you should see that the 'No Persisted Data' box in your channel updates with data sent by our BME280 and TSL2561 sensors! You may need to refresh the page in your browser a few times if not. Now press **CTRL+C** to stop the code running.

“ Beebotte has a very simple dashboard feature and allows you to create a wide range of data visualisations ”

08 What the code does

Hopefully by now you should have a few readings for temperature, humidity, pressure, and luminosity in your Beebotte channel. If not, you may need to re-check everything, including the connections to your sensors. The code you downloaded to your Raspberry Pi does a few things:

09 Build your dashboard

Now that we have data being sent to Beebotte every 15 minutes, we can start building our home monitoring dashboard so that we can see the data in graphical form. Thankfully, Beebotte has a very simple dashboard feature and allows you to create a wide range of data visualisations with a few clicks. Why not create a timeline graph that shows the temperature readings, or a gauge meter to show the last luminosity reading? If you have multiple Raspberry Pi computers with the same sensors and code running, why not combine them to compare? For some inspiration, check out our example dashboard by visiting magpi.cc/beebottedash.

10 Public vs private dashboards


Privacy is an important issue and some people may not want to share their data publicly. Fortunately, Beebotte allows you to choose whether to make your channels and/or dashboards public or private, depending on your preferences. Our dashboard has been made public so that anybody with the link can view it, but unchecking the 'public' box when creating or editing the

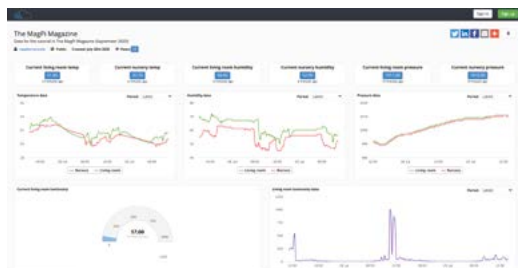
dashboard changes this. Unless you are storing and visualising sensitive data, making your dashboard public should be fine – and it means you can share with friends or family easily – but keep this in mind when creating yours.

11 Add more sensors

In our example, we have chosen the BME280 and TSL2561 sensors because they are affordable and readily available, plus have the benefit of being widely supported on Raspberry Pi. However, you can also add other sensors and can send the data to Beebotte very easily. For example, the Enviro board from Pimoroni works just as well, as does a DHT11 temperature and humidity sensor. Beebotte has easy-to-follow documentation and tutorials, so why not experiment with a variety of sensors? For example, what about recording ambient sound levels with a MEMS microphone sensor to check for noise pollution?

12 Multi-room sensing

Now you know how to monitor the temperature, humidity, pressure, and luminosity in one area/room, why not ramp things up and add another Raspberry Pi with sensors in another room? With Raspberry Pi Zero being small and affordable, it makes a perfect choice for hiding away in other rooms in your home, and with Beebotte you can easily monitor and compare data for different rooms in one place. Just create a new channel for each Raspberry Pi, but use the dashboard to combine the data in one super graph – it's also great for air conditioning wars in the office! 



▲ Create your own dashboards in Beebotte using the data collected in your channel(s). Why not combine multiple channels from various sensors for a comparison?

beebotte.py

► Language: Python

DOWNLOAD
THE FULL CODE:

 magpi.cc/beebotte

```
001. #!/usr/bin/env python
002.
003. import bme280
004. import time
005. import smbus
006. import sys
007. from beebotte import *
008.
009. # Replace CHANNEL_TOKEN with that of your Beebotte channel
    and YOUR_CHANNEL_NAME with the name you give your Beebotte
    channel
010. bbt = BBT(token = 'CHANNEL_TOKEN')
011. chanName = "YOUR_CHANNEL_NAME"
012.
013. # Set the time to wait between sending data to Beebotte.
    Avoid lower values due to API limitations. Default is 900
    seconds (15 mins).
014. period = 900
015.
016. # These resources needed to be added to your Beebotte channel
    - make sure the channel resource names match!
017. temperature_resource = Resource(bbt, chanName, 'temperature')
018. humidity_resource = Resource(bbt, chanName, 'humidity')
019. pressure_resource = Resource(bbt, chanName, 'pressure')
020. luminosity_resource = Resource(bbt, chanName, 'luminosity')
021.
022. # Set up the I2C bus for the TSL2561 sensor
023. bus = smbus.SMBus(1)
024.
025. # Read the luminosity data from the TSL2561 sensor
026. def get_luminosity():
027.     bus.write_byte_data(0x39, 0x00 | 0x80, 0x03)
028.     bus.write_byte_data(0x39, 0x01 | 0x80, 0x02)
029.     time.sleep(0.5)
030.     full_data = bus.read_i2c_block_data(0x39, 0x0C | 0x80, 2)
031.     ir_data = bus.read_i2c_block_data(0x39, 0x0E | 0x80, 2)
032.     full_spectrum = full_data[1] * 256 + full_data[0]
033.     infrared = ir_data[1] * 256 + ir_data[0]
034.     visible = full_spectrum - infrared
035.     return visible
036.
037. # The main part. Get the data from the sensors, then send it
    to your Beebotte channel.
038. def run():
039.     while True:
040.         temperature, pressure, humidity = bme280.readBME280All()
041.         luminosity = get_luminosity()
042.         if temperature is not None and humidity is not None and
            pressure is not None and luminosity is not None:
043.             try:
044.                 temperature_resource.write(round(temperature,1))
045.                 humidity_resource.write(round(humidity,0))
046.                 pressure_resource.write(round(pressure,0))
047.                 luminosity_resource.write(round(luminosity,0))
048.             except Exception:
049.                 print ("Error while writing to Beebotte")
050.         else:
051.             print ("Failed to get readings. Try again!")
052.
053.         time.sleep(period)
054.
055. run()
```



Connect Raspberry Pi 4 to iPad Pro with a USB-C cable



Lucy Hattersley

Lucy is the editor of *The MagPi* and knows more than is healthy about Apple. This project brings together both worlds.

magpi.cc

Our Raspberry Pi 4 iPad Pro setup adds programming power to Apple's closed-source device. Both data and power are shared using the USB-C cable

In many ways, Apple's iPad Pro is the polar opposite of Raspberry Pi. While both are superbly engineered devices, Raspberry Pi is a great programmable computer while iPad Pro is a highly closed system.

The two make a great tag-team combination, though. Launched in 2018, the iPad Pro comes with a USB-C connection, as does Raspberry Pi 4.

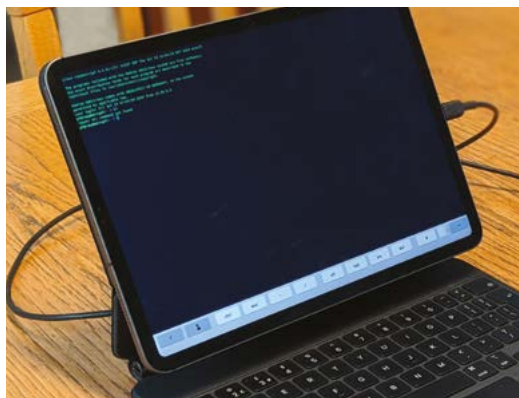
With a little tweaking, you can enable Raspberry Pi 4 data sharing via the USB-C cable and set up a static IP address. Get it to set everything up at boot time and you can power and remote-control your Raspberry Pi with just a single USB-C cable.

01 Clean install

We start with a fresh installation of Raspberry Pi OS (32-bit) flashed to a microSD card using Raspberry Pi Imager (magpi.cc/imager).

Set up Raspberry Pi with a monitor, keyboard, and mouse (we won't need these by the end of the tutorial). Power up and go through the

“ Make sure you change the default password during setup ”



▲ An iPad app called Termius is used to SSH into Raspberry Pi and control it from the command line

setup process. Make sure you change the default password during setup.

Open a Terminal window and make sure everything is up to date with:

```
sudo apt update
sudo apt full-upgrade
sudo reboot
```

02 Turn on SSH and VNC

Click the Raspberry Pi menu icon and choose Preferences > Raspberry Pi Configuration. Click Interfaces and set both SSH and VNC to Enabled. Click OK to close the Raspberry Pi Configuration tool.

03 Adjust config

Now you need to adjust the resolution and ensure that `dtoverlay=dwc2` is at the end of your `config.txt` file.

```
sudo nano /boot/config.txt
```

Uncomment both framebuffer lines (remove the # before them) and change the resolution to 1024×768.

```
framebuffer_width=1024
framebuffer_height=768
```

Head to the end of the `config.txt` file and ensure that `dtoverlay=dwc2` is present:

```
[all]
dtoverlay=dwc2
```

Save and close the `config.txt` file with **CTRL+O**, then **CTRL+X**.

04 Adjust cmdline

Open the `cmdline.txt` file:

```
sudo nano /boot/cmdline.txt
```

Add a new line below `console=serial0, ...` and add the following:

```
modules-load=dwc2
```

05 Adjust modules

Next, we adjust the `modules` file:

```
sudo nano /etc/modules
```

Add this command to the end:

```
libcomposite
```

Save and close.

06 Fix the IP address

Now we need to prevent Raspberry Pi from choosing its internet address. Edit the `dhcpcd.conf` file:

```
sudo nano /etc/dhcpcd.conf
```

Add this to the end of the file:

```
denyinterfaces usb0
```

You'll Need

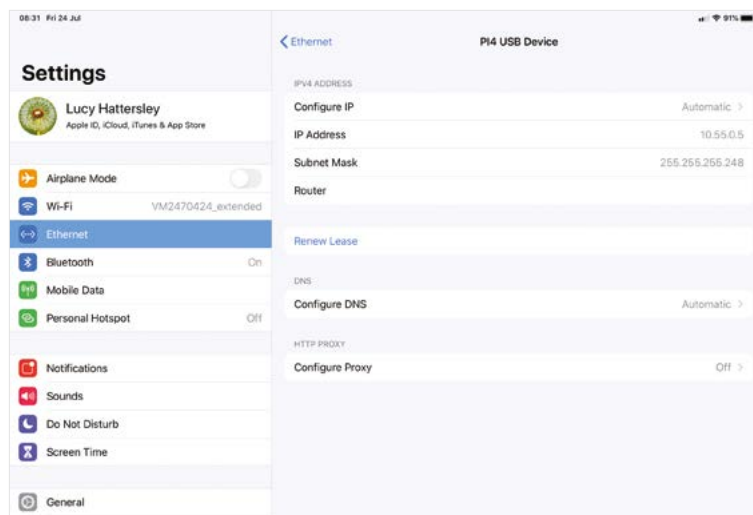
- iPad Pro
- Raspberry Pi 4
- Compatible USB-C cable
- Termius app
- VNC Connect or Screens app



Warning! Security

Turning on SSH and VNC opens ports on your Raspberry Pi. Make sure you change the default password.

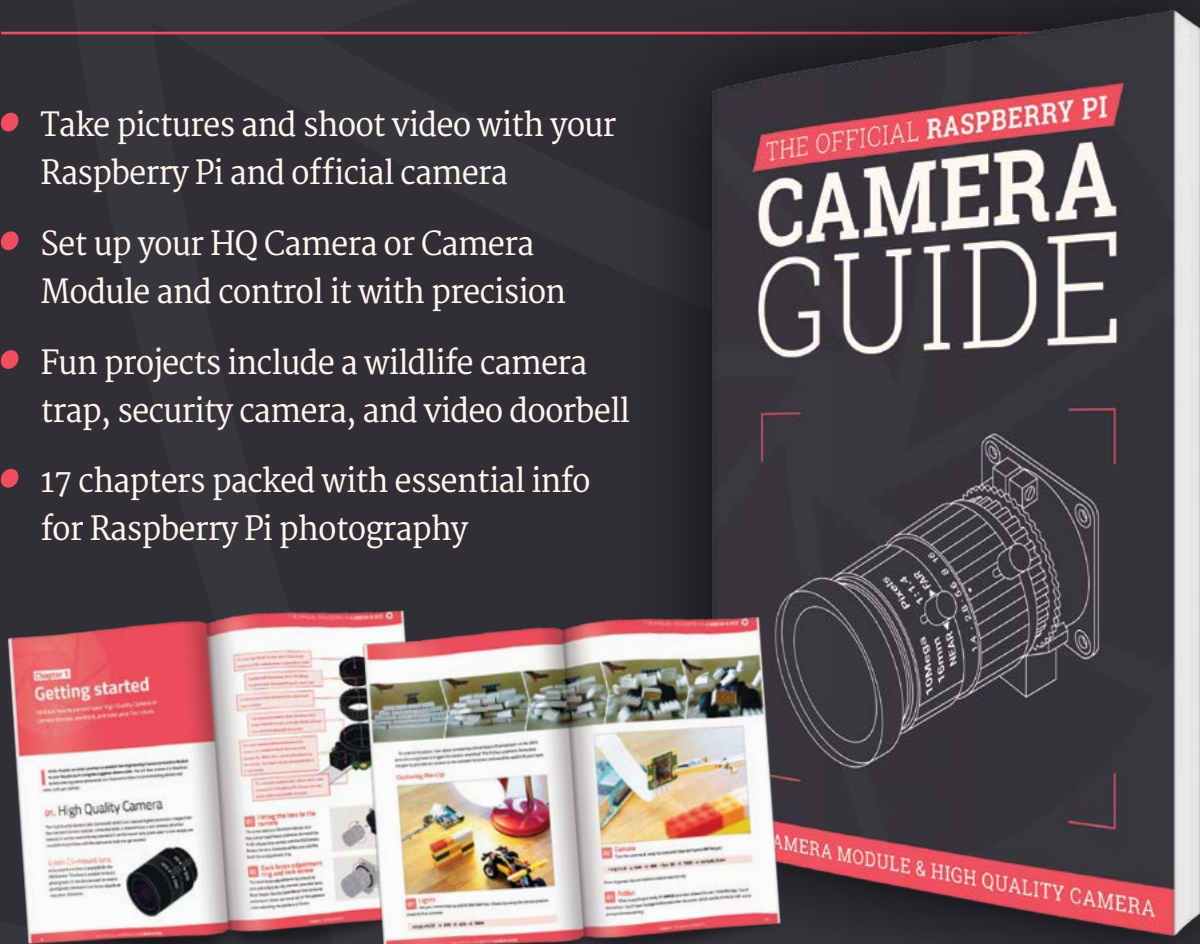
magpi.cc/security



▼ The Raspberry Pi Ethernet settings displayed in iPadOS

THE OFFICIAL RASPBERRY PI CAMERA GUIDE

- Take pictures and shoot video with your Raspberry Pi and official camera
- Set up your HQ Camera or Camera Module and control it with precision
- Fun projects include a wildlife camera trap, security camera, and video doorbell
- 17 chapters packed with essential info for Raspberry Pi photography



Buy online: magpi.cc/cameraguide

Top Tip



Esc key

Locate the Hardware Keyboard settings in iPadOS and set the Globe Key to Escape.

07 Choose an IP range

Install dnsmasq:

```
sudo apt install dnsmasq -y
```

Now create a **usb** file:

```
sudo nano /etc/dnsmasq.d/usb
```

And place the following script in it:

```
interface=usb0
dhcp-range=10.55.0.2,10.55.0.6,255.255.255.248,1h
dhcp-option=3
leasefile-ro
```

08 Choose an address

Now it's time to pick a static IP address.

This can be used to connect to Raspberry Pi from the iPad Pro.

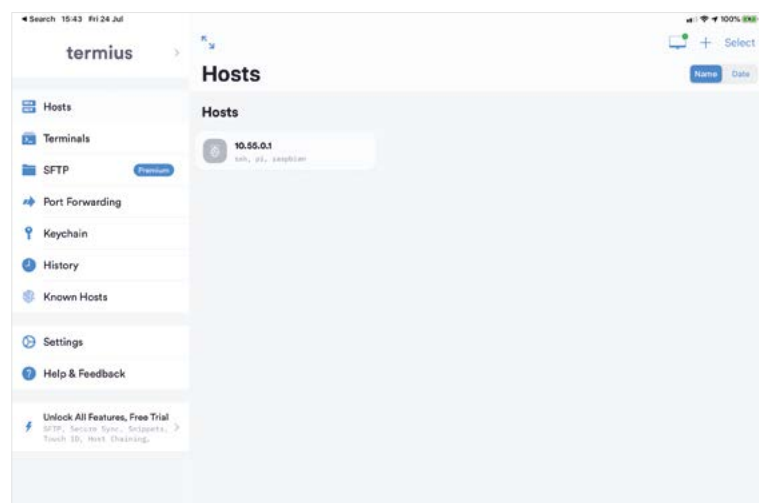
```
sudo nano /etc/network/interfaces.d/usb0
```

And add the following script:

```
auto usb0
allow-hotplug usb0
iface usb0 inet static
address 10.55.0.1
netmask 255.255.255.248
```

Save and close the file. Our IP address is 10.55.0.1. We will use this (or raspberrypi.local) to SSH and VNC into Raspberry Pi.

▼ The Raspberry Pi shows up as a USB Ethernet connection in iPad settings



09 Run the script

Now we need to run the **usb.sh** script on our Raspberry Pi. You can add the code manually, which is better for understanding.

```
sudo nano /root/usb.sh
```

Or quickly cut-and-paste it from Ben Hardill's website: magpi.cc/hardill. If you want to cut and paste, it's better to use Text Editor:

```
sudo mousepad /root/usb.sh
```

...and fill out the code from **usb.sh**. Save and exit. Now make the **usb.sh** file executable with:

```
sudo chmod +x /root/usb.sh
```

“ It's a good idea to keep the screen connected while you test everything ”

10 Run the script at launch

We're going to ensure the **usb.sh** script runs when we boot up Raspberry Pi, to ensure we can always access it. Use crontab:

```
sudo crontab -e
```

The first time we do this, it'll ask us to select an editor; we're going to use nano: enter 1.

Now add the following line to the end of crontab.

```
@reboot bash /root/usb.sh
```

Save and exit with **CTRL+O** and **CTRL+X**.

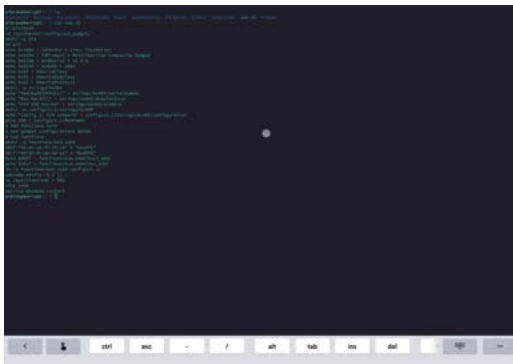
11 Swap to iPad

Shut down Raspberry Pi:

```
sudo shutdown -h now
```

Disconnect Raspberry Pi from the power source and connect the USB-C cable to your iPad Pro. It's a good idea to keep the screen connected while you test everything is working.

Not all USB-C cables work, and we didn't get it to work with the Apple USB-C cable. We're using an Anker cable. It should reboot as normal (with the screen resolution size from Step 4).



▲ Termius is an SSH app that enables you to control your Raspberry Pi from the command line

12 Check the connection

If all is working correctly, you will see your Raspberry Pi as a new Ethernet connection in iPadOS's Settings app.

You should see a new option, Ethernet, beneath WiFi. Click it to see 'Pi4 USB Device'. Click Pi4 and you will see the Pi4 USB Device settings.

You're now ready to use SSH and VNC to access Raspberry Pi from your iPad Pro.

13 VNC Viewer

VNC Viewer is available from the App Store. Open it and click Add. Enter 10.55.0.1 in the address field and give it a name (we called ours Raspberry Pi 4 iPad Pro). Click Save and click Connect.

Enter 'pi' in the Username field and your password, then click Continue. You will now see your desktop in VNC on iPad Pro.

14 Screens

Screens (magpi.cc/screensios) is a better VNC solution because you can point and click the pointer directly on the screen (instead of pushing a virtual pointer around). You'll need to adjust your VNC Server settings to use it.

Open VNC Server on Raspberry Pi (click on the VNC Server icon) and click on the menu icon, then choose Options. Change the authentication from UNIX password to VNC.

Click Users & Permissions and click Password. Enter a custom VNC password in both fields and click OK.

Back in Screens, click the '+' icon and choose New Screen. Click Custom and enter 10.55.0.1 in the Address field. Change the operating system to Raspberry Pi, and enter 'pi' in the Username field. Click Done. Click on the screen icon to start a connection.

usb.sh

> Language: **Bash**

**DOWNLOAD
THE FULL CODE:**



magpi.cc/hardill

```
001. #!/bin/bash
002. cd /sys/kernel/config/usb_gadget/
003. mkdir -p pi4
004. cd pi4
005. echo 0x1d6b > idVendor # Linux Foundation
006. echo 0x0104 > idProduct # Multifunction Composite Gadget
007. echo 0x0100 > bcdDevice # v1.0.0
008. echo 0x0200 > bcdUSB # USB2
009. echo 0xEF > bDeviceClass
010. echo 0x02 > bDeviceSubClass
011. echo 0x01 > bDeviceProtocol
012. mkdir -p strings/0x409
013. echo "fedcba9876543211" > strings/0x409/serialnumber
014. echo "Ben Hardill" > strings/0x409/manufacturer
015. echo "PI4 USB Device" > strings/0x409/product
016. mkdir -p configs/c.1/strings/0x409
017. echo "Config 1: ECM network" >
    configs/c.1/strings/0x409/configuration
018. echo 250 > configs/c.1/MaxPower
019. # Add functions here
020. # see gadget configurations below
021. # End functions
022. mkdir -p functions/ecm.usb0
023. HOST="00:dc:c8:f7:75:14" # "HostPC"
024. SELF="00:dd:dc:eb:6d:a1" # "BadUSB"
025. echo $HOST > functions/ecm.usb0/host_addr
026. echo $SELF > functions/ecm.usb0/dev_addr
027. ln -s functions/ecm.usb0 configs/c.1/
028. udevadm settle -t 5 || :
029. ls /sys/class/udc > UDC
030. ifup usb0
031. service dnsmasq restart
```

15 Termius for SSH

There are many SSH apps from iPadOS, and we find Termius (magpi.cc/termius) to be our favourite. Open Termius and click the Add icon ('+') and choose New Host. Enter 10.55.0.1 in the Hostname field. Enter 'pi' in the Username field. Now click Save and click on the new Host to connect to it.

There are other ways to interact with Raspberry Pi from an iPad, but SSH and VNC will be enough to get you started.

Raspberry Pi makes the ideal companion for iPad Pro, enabling you to program, code, and learn with the convenience of iPad Pro's long battery life and superb display. [M](#)

Top Tip



Type # in
Screens

Open Settings in Screens and disable the 'Use option as Meta key' option to type a # key on an iPad Pro UK keyboard.

HackSpace

TECHNOLOGY IN YOUR HANDS

THE **MAGAZINE**
FOR THE **MODERN MAKER**



ISSUE **#34**
OUT NOW

hsmag.cc



Available on the
App Store



GET IT ON
Google Play

Tap-an-LED: the hardware



Mike Cook

Veteran magazine author from the old days, writer of the Body Build series, plus co-author of *Raspberry Pi for Dummies*, *Raspberry Pi Projects*, and *Raspberry Pi Projects for Dummies*.

magpi.cc/mikecook

Tap-an-LED brings you a touch interface on a 16×8 multicoloured LED matrix, for emulating a Monome, generating new user interfaces, and playing games

01 What's the big idea?

The idea behind this project is simple: take a glass-resistive touchscreen and put it over an LED matrix and use the matrix as if it had illuminated push-buttons. In practice, however, we found it harder to implement than we would have liked. An illuminated push-button matrix is at the heart of many user interfaces for music, like the Monome, Tenori-on, and Launchpad. However, these are expensive, mainly due to the cost of a lot of illuminated push-buttons. By combining a touchscreen and LED matrix, we bring the cost right down.

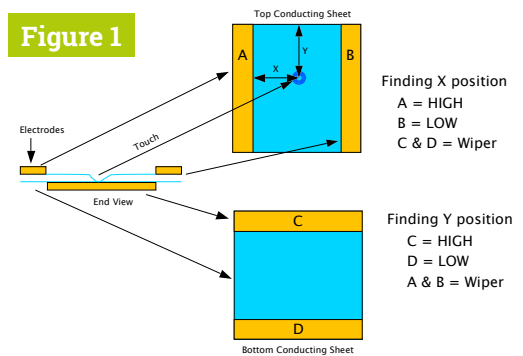
02 How do we do this?

Touchscreens are designed to provide an overlay to a display screen, to turn them into a touch-sensitive display. Display screens are relatively expensive, and they tend to only have a very limited time when they are available. But combining a touch overlay to an LED matrix gives you a low-cost solution. A resistive touchscreen is made by coating a glass sheet with two layers of conducting indium tin oxide (ITO). These are separated by an air (or inert gas) gap and when the user touches the screen, it deforms the top layer so it touches the bottom layer.

03 How do touchscreens work?

The trick here is that the conducting coating has a resistance that can easily be measured, and so the touch can look just like a potentiometer. With a rotary potentiometer, you can tell the position of the rotation shaft by measuring the voltage at the wiper; if it is half the voltage applied across the potentiometer, then the shaft is half

Figure 1



▲ Figure 1 How a resistive touchscreen works

way round. What the touchscreen does is the same, only in two dimensions (Figure 1). The electrodes are low-resistance conducting strips and by making each one high and low on one surface, and joining the two on the other surface to make a wiper, the X-Y location can be found.

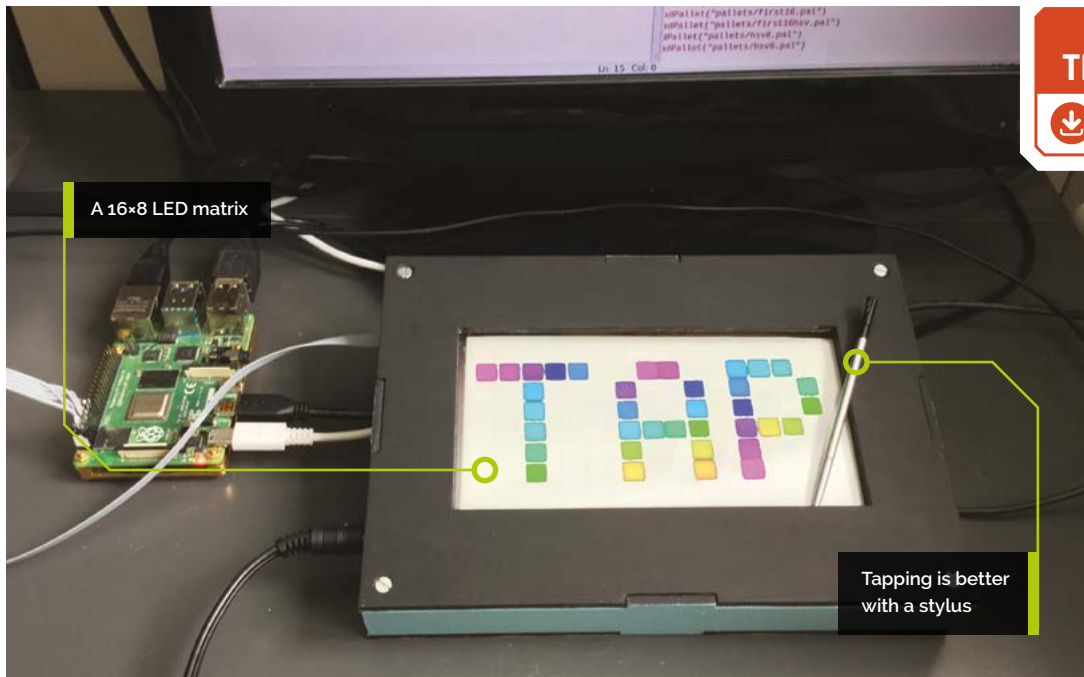
“Combining a touch overlay to an LED matrix gives you a low-cost solution”

04 One chip to do it all

This trick of switching the voltages on the four electrodes could be done with a handful of parts, but it is more convenient to integrate these functions into a single chip. The chip needs to be able to switch voltages on the electrodes, and steer the wiper into an A/D converter. There are a few chips that will do this, but one that has been conveniently placed on a breakout board is the STMPE610. This chip contains both an I²C and SPI interface; we chose to use the I²C because it uses fewer wires and was fast enough for our application here.

You'll Need

- 2 × 8×8 WS2812 LED matrix
magpi.cc/ws2812matrix
- 7.1" Resistive touchscreen sensor
magpi.cc/touchoverlay
- STMPE 610 interface board
magpi.cc/stmpe610



**DOWNLOAD
THE FULL CODE:**

 magpi.cc/pibakery



Alert! Drill safety

Please use safety goggles and a desk clamp when drilling.

magpi.cc/drillsafety

Top Tip 

Getting the matrix right

We marked what was to be the bottom left-hand corner of both matrices with a blob of white correcting fluid. This is the corner where the 'data in' is marked on the underside.

05 The LED matrix

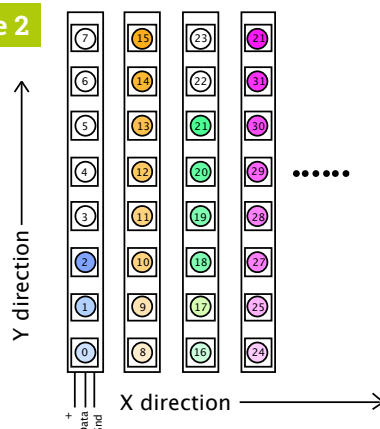
These days an 8×8 matrix made out of WS2812 multicoloured LEDs is quite cheap. We used two of these to form a 16×8 LED matrix. The final matrix size was governed by how many boards would fit under the touchscreen. There are two ways you can wire a matrix: either a raster or a serpentine raster. A raster is where there are just lines of LEDs for consecutive numbers; a serpentine raster is where adjacent lines go in different directions, like a snake.

be wired. What we have implemented here is an ascending column-based raster matrix. This is where we start off, with the first LEDs being on a column going up; when the numbers reach the top, it starts again on the next column along. The LED numbering is shown in **Figure 2** for the first four columns of the matrix. We will use a very simple maths routine to convert any required X-Y location into an LED number.

06 Arrangement we used

Couple that with the fact you can have four different orientations of the square matrix, and it gives you eight different ways a matrix can

Figure 2



07 Schematic

The LED matrix requires a 5V signal and supply voltage, so we need to convert Raspberry Pi's 3V3 signals into 5V. We also found that our official Raspberry Pi power supply was quite noisy. **Figure 3** shows a 2V burst of noise. This meant

◀ **Figure 2** LED numbering on an ascending column raster matrix

▼ **Figure 3** Noise spike on the supply voltage to the STMPE610

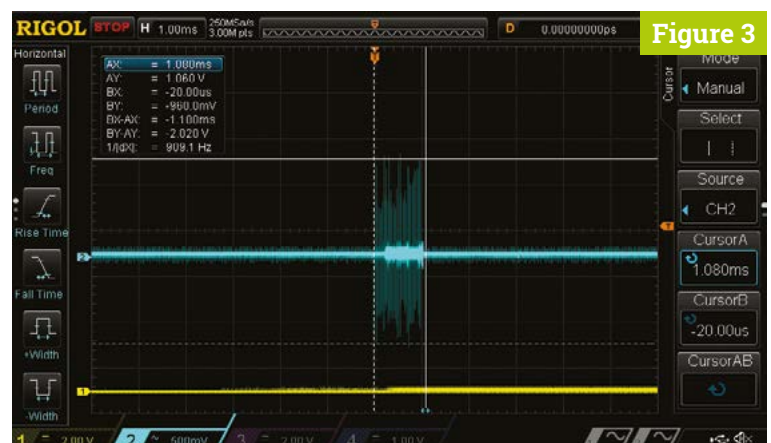
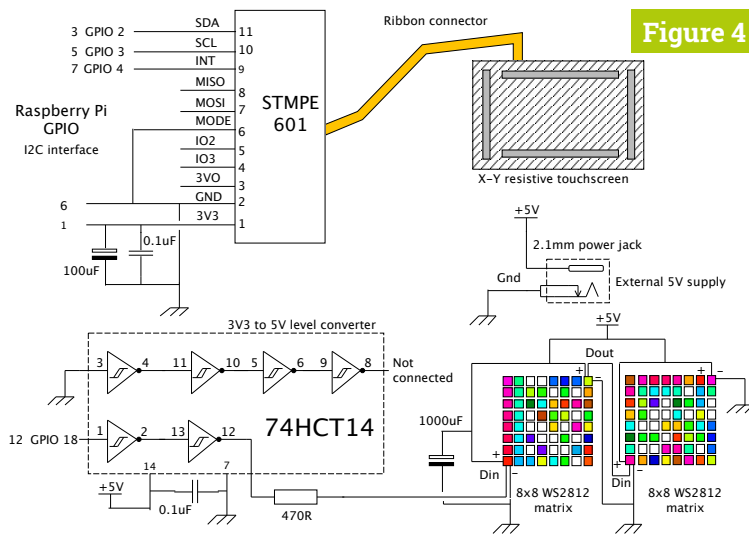


Figure 3



▲ **Figure 4** Schematic of the Tap-an-LED project

the readings we got back were very erratic, so we added some smoothing on the input to the STMPE610 with a small capacitor board. With the potential current for the project being very high, we decided to power the matrix only with an external supply. The full schematic of our project is shown in **Figure 4**.

Top Tip



Ventilation

In order to keep things cool, you might need some ventilation. We drilled six 4mm holes in the base at the right-hand side and the same in the left-hand side. We fixed small rubber feet on the base to create a through-draught.

► **Figure 5** Lining up the matrix boards with the light baffles

08 Get the spacing right

A parts list and case drawings can be found on our GitHub page. First, position the left-hand matrix board, and drill 3mm through the four corner holes into the base. Temporarily attach the first matrix board to the base. The second matrix board should not be fitted flush to the first, but must be separated from the first so that the LED spacing is the same over the two boards. This is best done by cutting one of the longer light baffle strips with the nut recess shapes in them and one of the short strips. Use these to position the gap between the two boards (**Figure 5**).

Figure 5

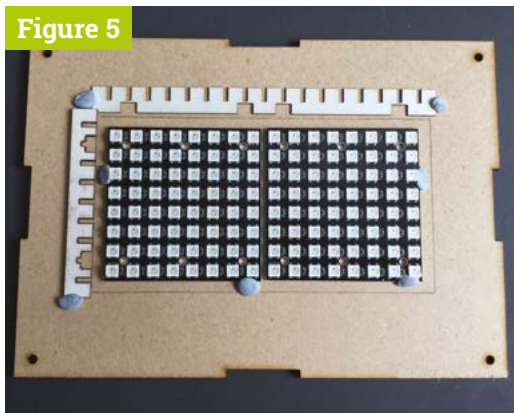
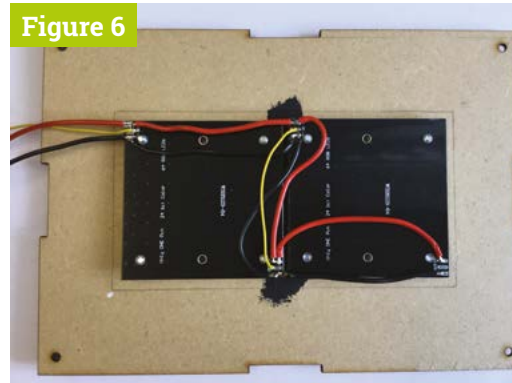


Figure 4

Figure 6



▲ **Figure 6** Wiring up all power and ground to the back of the matrix boards

09 Wire and fix the matrix boards

Place the two matrix boards upside down and push some screws through to fix them in place. Wire all the + and - supply lines together with 0.5 mm² stranded insulated wire, as shown in **Figure 6**. Make sure, when you turn them over, that the two marked corners are on the lower left-hand side. Mount the two matrix boards using 8mm countersunk screws on 5mm nylon spacers, and fix with a nylon M3 nut. Cut out the rest of the light baffle strips from 1.3mm white cardboard mounting board and assemble over the LED matrix.

10 Top of the box

Cut out the top of the box and position the touchscreen in the centre. Lay some 3mm strips of wood alongside the touchscreen and glue them in place. Next, cut two strips of 2mm foam for the top and bottom of the touchscreen to rest against. Tape across the corners with insulating tape, just to keep the touchscreen in place during assembly (**Figure 7**). Make the matrix driver board using the layout in **Figure 8** and fix to the bottom left-hand side of the matrix. Solder a strip of right-angled pins to the STMPE610 board (**Figure 9**) – don't use the vertical ones that came with it.

11 Box assembly

Drill two 2.5mm holes in the base to match the STMPE610 board, and mount it at the top left-hand side of the matrix. Fix an M3 24mm brass spacer to each corner of the base. We didn't have any of these, so used a 12mm male-female spacer screwed into a 12mm female-female spacer.

Cut out the four sides of the box and glue them together on their edges only; we used transparent tape to stop the glue running down the sides and sticking to the base. Screw the top to the spacers, apply glue to the sides, and tighten up.

“ A parts list and case drawings can be found on our GitHub page ”

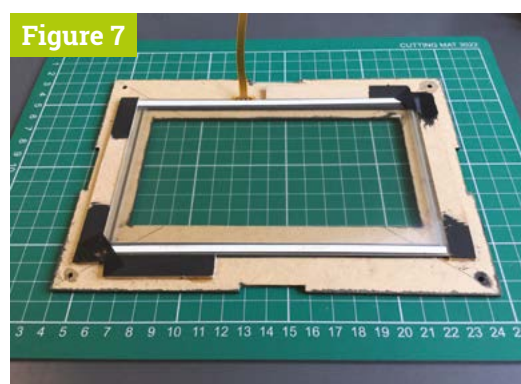
12 Final assembly

Make up the box / Raspberry Pi lead from 6-way ribbon cable, as shown in **Figure 10**. It is best to use a diffuser at the top of the baffle – we used a 0.5mm styrene sheet; just tack it with glue on the corners to keep it in place. Wire up the power jack through the hole in the side, remove the protective plastic sheets from the touch sensor, plug it into the STMPE610 board with the black side up, and screw the whole assembly together. Don't over-tighten the screws, or else there is a danger of cracking the glass.

13 Testing

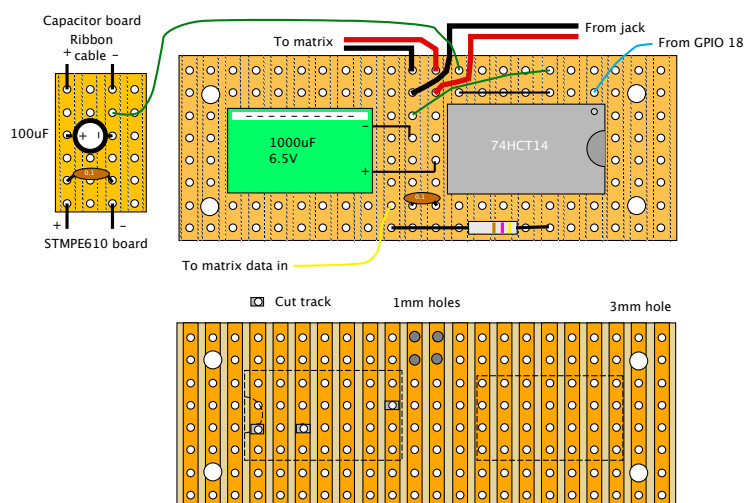
It is best to test each bit as you go along. We have some test software on our GitHub page to test out the matrix and the touch sensor board. We found it best to use a stylus to do the tapping – get one of the sharp-pointed ones, not the colourful soft rubber-ended ones that are used for capacitive touchscreens.

Next month, in part two, we will look at how to calibrate your touch sensor, and show you some of the things you can make with this board. **M**



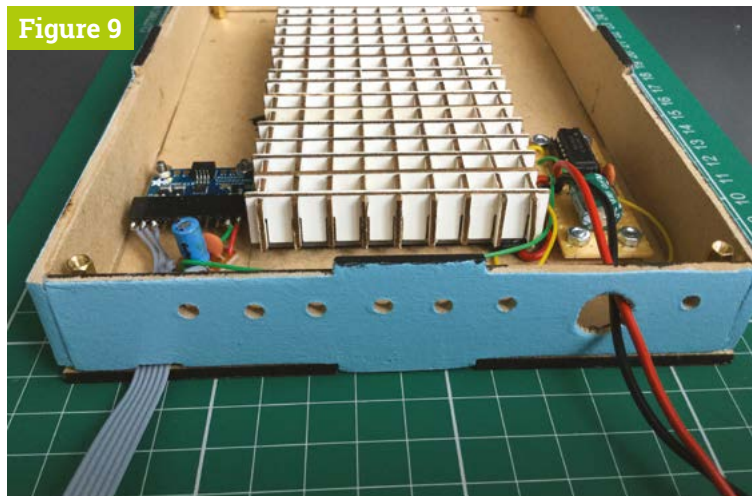
▲ **Figure 7** Fitting the glass touchscreen to the top of the box

Figure 8



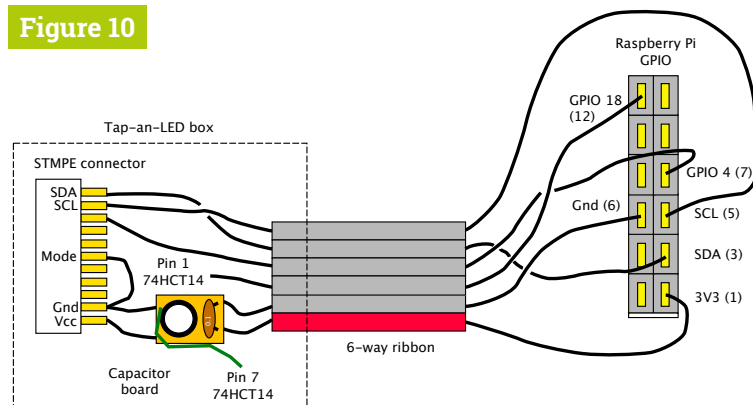
▲ **Figure 8** Physical layout of the Tap-an-LED schematic

Figure 9

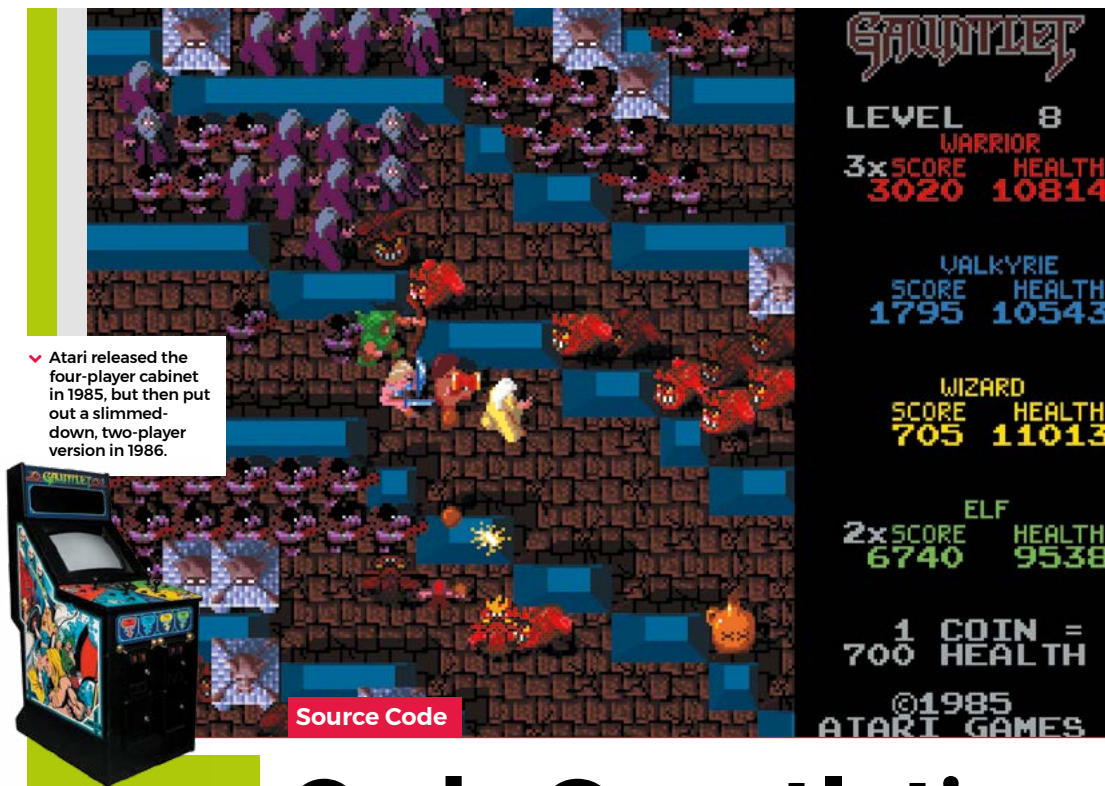


▲ **Figure 9** Fitting the two boards either side of the light baffles

Figure 10



▲ **Figure 10** Wiring the 6-way ribbon cable to Raspberry Pi's GPIO connector



✓ Atari released the four-player cabinet in 1985, but then put out a slimmed-down, two-player version in 1986.

Source Code



Wireframe

This tutorial first appeared in Wireframe, our sister magazine that lifts the lid on the world of video games. Every issue includes tutorials and in-depth interviews, along with news and reviews of the latest indie and triple-A games.

To find out more, visit their website at wfmag.cc.

Check out their subscription offers at wfmag.cc/subscribe.



AUTHOR
MARK VANSTONE

Code Gauntlet's four-player co-op

Four players dungeon crawling at once? Mark shows you how

Atari's *Gauntlet* was an eye-catching game, not least because it allowed four people to explore its dungeons together. Each player could choose one of four characters, each with its own abilities – there was a warrior, a Valkyrie, a wizard, and an elf – and surviving each dungeon required slaughtering enemies and the constant gathering of food, potions, and keys that unlocked doors and exits.

Designed by Ed Logg, and loosely based on the tabletop RPG *Dungeons & Dragons*, as well as John Palevich's 1983 dungeon crawler, *Dandy*, *Gauntlet* was a big success. It was ported to most of the popular home systems at the time, and Atari released a sequel arcade machine, *Gauntlet II*, in 1986.

Atari's original arcade machine featured four joysticks, but our example will mix keyboard controls and gamepad inputs. Before we deal with the movement, we'll need some characters and dungeon graphics. For this example, we can make our dungeon

from a large bitmap image and use a collision map to prevent our characters from clipping through walls. We'll also need graphics for the characters moving in eight different directions. Each direction has three frames of walking animation, which makes a total of 24 frames per character. We can use a Pygame Zero Actor object for each character and add a few extra properties to keep track of direction and the current animation frame. If we put the character Actors in a list, we can loop through the list to check for collisions, move the player, or draw them to the screen.

We now test input devices for movement controls using the built-in Pygame keyboard object to test if keys are pressed. For example, `keyboard.left` will return `True` if the left arrow key is being held down. We can use the arrow keys for one player and the **WASD** keys for the other keyboard player. If we register x and y movements separately, then if two keys are pressed – for example, up and left – we can read that as a diagonal movement. In this way, we can get all eight directions of movement from just four keys.

For joystick or gamepad movement, we need to import the joystick module from Pygame. This provides us with methods to count the number of joystick or gamepad devices that are attached to the computer, and then initialise them for input. When we check for input from these devices, we just need to get the x-axis value and the y-axis value and then make it into an integer. Joysticks and gamepads should return a number between -1 and 1 on each axis, so if we round that number, we will get the movement value we need.

We can work out the direction (and the image we need to use) of the character with a small lookup table of x and y values and translate that to a frame number cycling through those three frames of animation as the character walks. Then all we need to do before we move the character is check they aren't going to collide with a wall or another character. And that's it – we now have a four-player control system. As for adding enemy spawners, loot, and keys – well, that's a subject for another time. ☺



Download
the code
from GitHub:
[wfmag.cc/
wfmag39](https://wfmag.cc/wfmag39)

Four-player movement in Python

Here's Mark's code for a *Gauntlet*-style four-player mechanic. You'll need to install Pygame Zero – see wfmag.cc/pgzero. To run the code, **cd** to its directory, then **pgzrun gauntlet.py**

```
import math
from pygame import image, Color, joystick

myChars = []
myDirs = [(0,1),(-1,1),(-1,0),(-1,-1),(0,-1),(1,-1),
(1,0),(1,1)]
collisionmap = image.load('images/collisionmap.png')
joystick.init()
joyin0 = joyin1 = False
if(joystick.get_count() > 0):
    joyin0 = joystick.Joystick(0)
    joyin0.init()
if(joystick.get_count() > 1):
    joyin1 = joystick.Joystick(1)
    joyin1.init()

def makeChar(name,x,y):
    c = len(myChars)
    myChars.append(Actor(name+"_1", (x, y)))
    myChars[c].name = name
    myChars[c].frame = myChars[c].movex = myChars[c].movey =
myChars[c].dir = 0

def draw():
    screen.blit("colourmap", (0,0))
    drawChars()

def drawChars():
    for c in range(len(myChars)):
        myChars[c].image = myChars[c].
name+"_"+str(((myChars[c].dir*3)+1)+math.floor(myChars[c].
frame/10))
        myChars[c].draw()

def update():
    checkInput()
    moveChars()

def checkInput():
    if keyboard.left: myChars[0].movex = -1
    if keyboard.right: myChars[0].movex = 1
    if keyboard.up: myChars[0].movey = -1
    if keyboard.down: myChars[0].movey = 1
    if keyboard.a: myChars[1].movex = -1
    if keyboard.d: myChars[1].movex = 1
    if keyboard.w: myChars[1].movey = -1
    if keyboard.s: myChars[1].movey = 1
    if joyin0:
        myChars[2].movex = round(joyin0.get_axis(0))
        myChars[2].movey = round(joyin0.get_axis(1))
    if joyin1:
        myChars[3].movex = round(joyin1.get_axis(0))
```

```
myChars[3].movey = round(joyin1.get_axis(1))
```

```
def moveChars():
    for c in range(len(myChars)):
        getCharDir(myChars[c])
        if myChars[c].movex or myChars[c].movey:
            myChars[c].frame += 1
            if myChars[c].frame >= 30: myChars[c].frame = 0
            testmove = (int(myChars[c].x + (myChars[c].movex
*20)),int(myChars[c].y + (myChars[c].movey *20)))
            if collisionmap.get_at(testmove) == Color('black')
and collideChars(c,testmove) == False:
                myChars[c].x += myChars[c].movex
                myChars[c].y += myChars[c].movey
                myChars[c].movex = 0
                myChars[c].movey = 0

def getCharDir(ch):
    for d in range(len(myDirs)):
        if myDirs[d] == (ch.movex,ch.movey):
            ch.dir = d

def collideChars(c,xy):
    for ch in range(len(myChars)):
        if myChars[ch].collidepoint(xy) and ch != c:
            return True
    return False

makeChar("warrior",60,60)
makeChar("valkyrie",500,450)
makeChar("wizard",460,180)
makeChar("elf",100,400)
```

Finding sprites

If you want to reconstruct a retro game like *Gauntlet*, you can often find sprite sheets online – these are bitmaps with all the frames of animation for a character on one sheet. A good source of these files is spriters-resource.com, where you'll find sprite sheets for a wide range of retro games. Some coding systems can use the sprite sheets as they are, but for this example we have cut them up into separate frames. You can do this with Sprite Sheet Slicer, available at wfmag.cc/slicer.

> Our four-player homage to the classic *Gauntlet* arcade game.



MAKE GAMES ON RASPBERRY PI

Start with and graduate from Scratch
before making fully-fledged games

Making games on Raspberry Pi is something a lot of people already know you can do, whether it's a quick and fun Scratch project or a little experiment in Pygame and/or Minecraft.

It's very easy to take it to the next step, thanks to excellent tools available for Raspberry Pi. It's a good idea not to jump into the deep end, though, so it's advisable to work your way up from Scratch cat to selling on Steam.

Write down some ideas, stock up on snacks, and let's get developing.

Make a game in Scratch

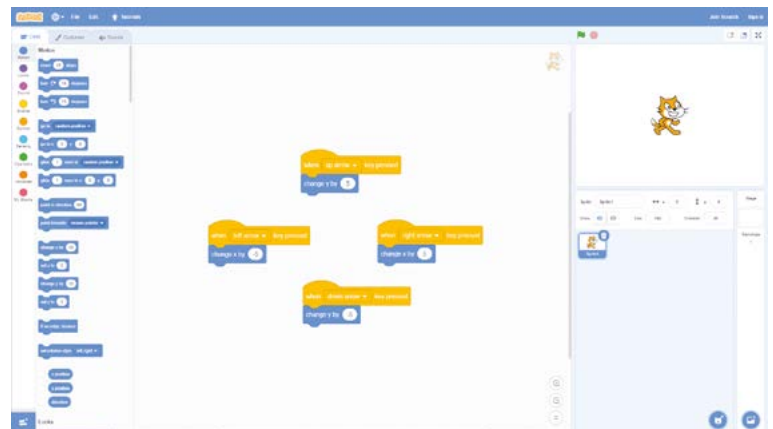
Start small and work your way up

Making games in Scratch is both easy and fun. Whether you're young or not-so-young, if you don't already have a good grasp of code, it's a great way to learn some of the basics and how to think about code in terms of games.

01 Prepare your game

From the Raspberry Pi menu, go to Programming and then Scratch 3. This will open up Scratch on your Raspberry Pi to start programming in. You'll be presented with a blank screen for your code, available blocks on one side, and the output of your code – the 'stage' – on the other. The stage is, by default, a white screen with the Scratch cat in the centre.

If you want, you can change the cat sprite by deleting it and adding a new one with the 'choose a sprite' option, and you can even add a background.



any key pressed' block so that it slots under it. Now by pressing any key, you can have Scratch cat move, albeit in the same direction.

▲ A simple Scratch program to move the cat sprite around

02 Object programming

Click on the cat and then concentrate on the code block part. Select the Events blocks category – where actions that will result in an event live – and look for the 'when X key is pressed' block. Drag it to the middle and change the key to 'any'. This block then looks for when any key is pressed, but it needs for something to happen when you press it.

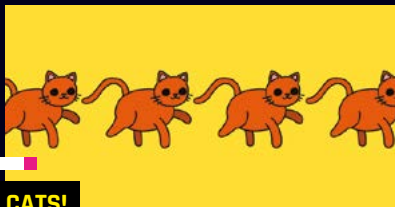
Select the Motion category and select the 'move X steps' block. Drag it over to go under the 'when

03 Understanding X and Y

We can have Scratch cat move in four directions (left, right, up, and down) by specifying what happens when each button is pressed. From the Motion category, select the 'change x' and 'change y' blocks. Add three more 'when x key pressed' blocks and assign them to the four arrow keys. To make Scratch cat move right and then left, you can set the 'change' number to a positive and negative one respectively – same with up and down.

You'll need to understand how things move on the X, Y, and later Z axes in 3D games to determine how characters move in the game space.

Complete game guides!



CATS!

Use your mouse to create a path over obstacles for cats to get to safety. This includes some basic AI and player interaction.

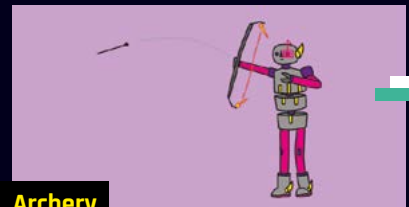
magpi.cc/scratchcats



Falling stars

A simple platformer where you need to collect stars to get a high score and avoid the enemy... which is a dog.

magpi.cc/fallingstars



Archery

Time your shot with the randomised crosshair. It will teach you about random movement, speeds, and timing.

magpi.cc/scratcharchery

Make a game in Ren'Py

Create games with basic Python

Install on Raspberry Pi

Head to renpy.org and go to the Downloads section. From there you'll need to download the tar.bz2 file of the SDK. Decompress it, and then download the Raspberry Pi Support file from the Additional Downloads section on the same page. Extract the files into the SDK directory and then run Ren'Py from that folder.

There are a few game engines out there that allow you to create simpler styles of game with some basic coding knowledge, and Ren'Py is one of them. With a basic level of Python knowledge, and strong Google skills, you'll be able to start making some games with the visual novel engine Ren'Py.

01 Create your characters

Start a new project and follow the instructions on setting it up. You can change this later, so don't worry too much.

From the launcher, you can select your game and what part to edit. In **script.py** you'll find your game. From here, we'll add characters using the Ren'Py specific language:

```
define m = Character('Me')
```

This simply creates the characters of Rob and the player (Me) so they can be used in a way you expect. We can also create an image for the Rob and Me characters by using:

```
image rob = "rob.png"
image me = "me.png"
```

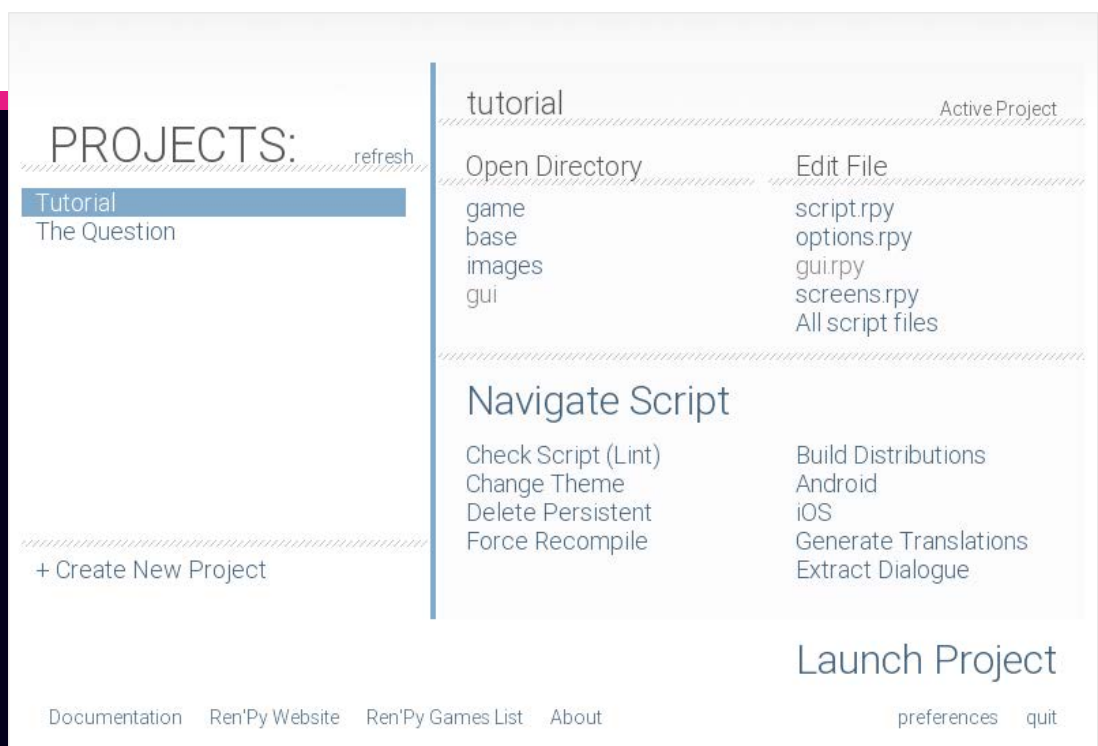
...with the files located in the **images** folder in your game directory.

“ You'll be able to start making some games with the visual novel engine Ren'Py **”**

```
define r = Character('Rob')
```

Make more

Head to the Ren'Py docs (magpi.cc/renpydocs) to learn much, much more



02 Give them a scene

In Ren'Py, specific game scenes are called 'labels' in the code. Think of a scene like a TV show or film – you're in a location, there's dialogue, and then we move on to the next scene. They work like this.

```
label start:

show rob

    r "Hi and welcome to The MagPi, the
official Raspberry Pi magazine."

    r "In this tutorial we'll teach you some
Ren'Py basics."

hide rob
show me

    m "Great, when do we start?"

hide me
show rob

    r "Right now!"

    "I let the corny joke go."
```

Text lines without a character attached are displayed to the player without one attached, so they can be used for thinking or out-of-story dialogue.

03 Choices

It's not much of a game if there's no interactivity. You can use menus to give your characters choices. These choices can then be used to move to different labels, or create a variable that will change the story down the line. You can create them with:

```
r "What would you like to do?"

menu:

    "Learn about Scratch.":
        jump scratch

    "Learn about PICO 8.":
        jump pico8
```

...where the `jump` command is used to move to labels with that name.

Ren'Py game types



Visual novel

An interactive story where you build relationships (romantic or otherwise) with different characters. There are usually lots of different 'routes' that let you play out the story in multiple ways.



Management sim

This kind of game is about managing and using resources to work towards a goal. Such as building a city with money and people, organising a football team, cooking for people, etc. Ren'Py can easily handle this.

Make a game in PICO-8

Go retro with this cult favourite game designer

The PICO-8 engine allows you to make games with a tiny resolution for very classic-looking games, albeit with many more colours than you could use in the days of the NES or Atari 2600. Some professionally released games have been made in it, so it's definitely worth looking at for something simpler.

Releasing your games

For non-Scratch games, we recommend using itch.io as a platform to release your work. It's free, and you can even charge for your game if you want.



The MagPi PICO-8 guide

Starting in issue 83, we ran a six-part series on making your own games in PICO-8 by Dan Howard. This included creating a character, levels, adding sounds and music, and finally polishing your game.

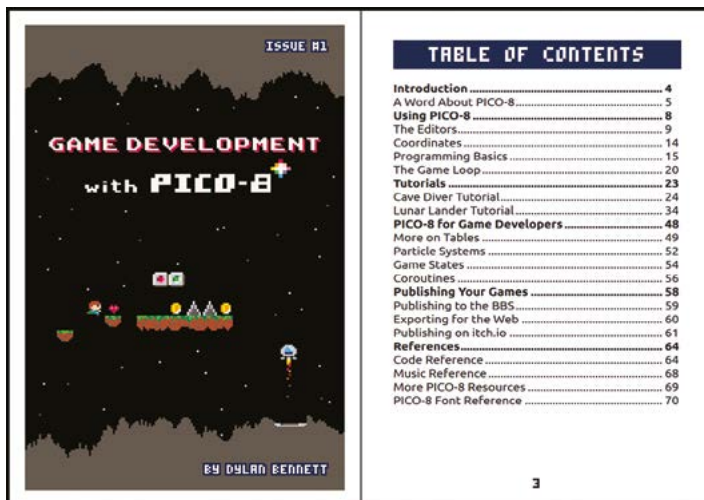
magpi.cc/83



PICO-8 Hero

This YouTube series goes through the basics and builds you up to making the game PICO-8 hero. It's great for people who learn a bit better visually, and is broken up into nice chunks so you're not having to learn too much at a time.

magpi.cc/pico8hero



Game development with PICO-8

This zine has loads of reference material for making games with PICO-8, including some basic tutorials on making simple games. It's a bit like *The MagPi*, but for PICO-8!

magpi.cc/pico8zine

PICO-8 games you can play now in your browser



Celeste

A PICO-8 version of the indie hit from a few years ago. It looks a lot simpler but is equally as tricky. It's from the creators of Tower Fall.

magpi.cc/celeste



Galactic Wars

A classic shoot-'em-up style game with a lot more punch and style than the R-Types of yore. It's recently had an extended version funded on Kickstarter as well.

magpi.cc/galacwars

Allocation

Allocation is known as a 'Metroidvana', a type of game where you go around collecting power-ups so you can access more of the level. This one has a really classic look straight from the early 1980s, albeit with modern sensibilities.

magpi.cc/allocation

MAGDOWNLOAD.ORG

FREE DOWNLOAD

LATEST MAGAZINES

HIGH QUALITY TRUE-PDF

MAGDOWNLOAD.ORG





Build Your Own FIRST-PERSON SHOOTER in Unity

Making a fast-paced 3D action game needn't be as daunting as it sounds. *Build Your Own First-Person Shooter in Unity* will take you step-by-step through the process of making *Zombie Panic*: a frenetic battle for survival inside a castle heaving with the undead.



IN THE PROCESS, YOU'LL DISCOVER HOW TO:

Set up and use the free
software you'll need

Create and texture 3D
character models

Make enemies that follow
and attack the player

Design a level with
locked doors and keys

Extend your game further,
with tips from experts

Available now: wfmag.cc/fps

CrowPi2

SPECS

DIMENSIONS:

291×190×46 mm;
1.3 kg

SCREEN:

11.6-inch
1920×1080 IPS

CAMERA:

2.0MP with
microphone

PERIPHERALS:

2.4GHz wireless
keyboard and
mouse

AUDIO:

Stereo speaker

► Elecrow ► magpi.cc/crowpi2 ► £231 / \$300

The exterior is a little rough, but inside beats a solid electronic heart. By **Lucy Hattersley**

CrowPi2 took us completely by surprise when it landed on our desk. We never tested the original CrowPi (\$229, magpi.cc/crowpi) which was a touchscreen and electronics kit.

This, however, is a huge step up. The touchscreen is replaced with a full IPS display, keyboard, and trackpad/mouse. Turning CrowPi2 into a fully-fledged laptop.

So much to discover

The real magic for CrowPi2 begins when you lift out the removable keyboard and take a look at the Development board beneath.

Here you'll find ["deep breath" – Ed]: a thumb joystick, segment display, relay, I/O LED, DHT11,

tilt switch, LCD, PIR motion sensor, sounds sensor, IR receiver, I/O interface, servo interface, step motor interface, 8×8 button matrix, a buzzer, vibration motor, touch sensor, 8×8 RGB LED matrix, light sensor, and an ultrasonic sensor.

Alongside all that is a 40-pin GPIO socket and breadboard with a GPIO toggle switch that you can use to connect/disconnect the components on the Development board.

The box contains a further 31 modules including an IR remote controller and receiver, stepper motor, servo, and DC motor; two game controllers and two 32 microSD cards (the spare is set up for RetroPie). And there are ten Minecraft paper cut-outs along with a user manual. Elecrow promises a Python and Scratch programming book to follow.

► Underneath the keyboard is a Development board, and the kit comes with an array of components





A custom version of Raspberry Pi OS boots into a friendly interface with links to projects, learning materials, and various other welcome features.

Start learning

The learning resources are excellent. There are 16 Scratch tutorials and 32 Python lessons. Both make use of the electronics components, having you build a flashing LED, automatic fan, tilt light – and, in the case of Python, working up to using sensors, RFID tags, voice recognition, and speech.

A further AI section has five Speech Recognition courses and seven Face Recognition lessons (using OpenCV and the built-in camera).

It was as we worked through these courses that the CrowPi2 came alive in our hands and we decided it was something special, worthy of our recommendation.

Design decisions

Objectively speaking, CrowPi2 as a laptop is far from perfect. Some of the design decisions are questionable. The keyboard is flimsy and bends, the key response isn't great. The trackpad is clunky. There are oddities like the camera USB cable that sticks out from the side and the lack of an internal battery.

Two things you'll need to get started aren't included: an AA battery for the mouse, and (less forgivably) a micro-USB cable to charge up the wireless keyboard. Given the sheer weight of kit included, this is an odd choice.

“ The learning resources are excellent ”

The fan whirs constantly. The unit is almost two inches thick.

What we have here, however, is enough value for money and high-quality learning resources to make us overlook all of those flaws.

The whole bundle

After a while, we began to admire CrowPi2 as a compact electronics learning kit rather than a laptop.

It's an all-in-one unit you can plug into a wall, put a kid (or willing-to-learn adult) in front of it, and let them get on with discovering how electronics components and computer code work.

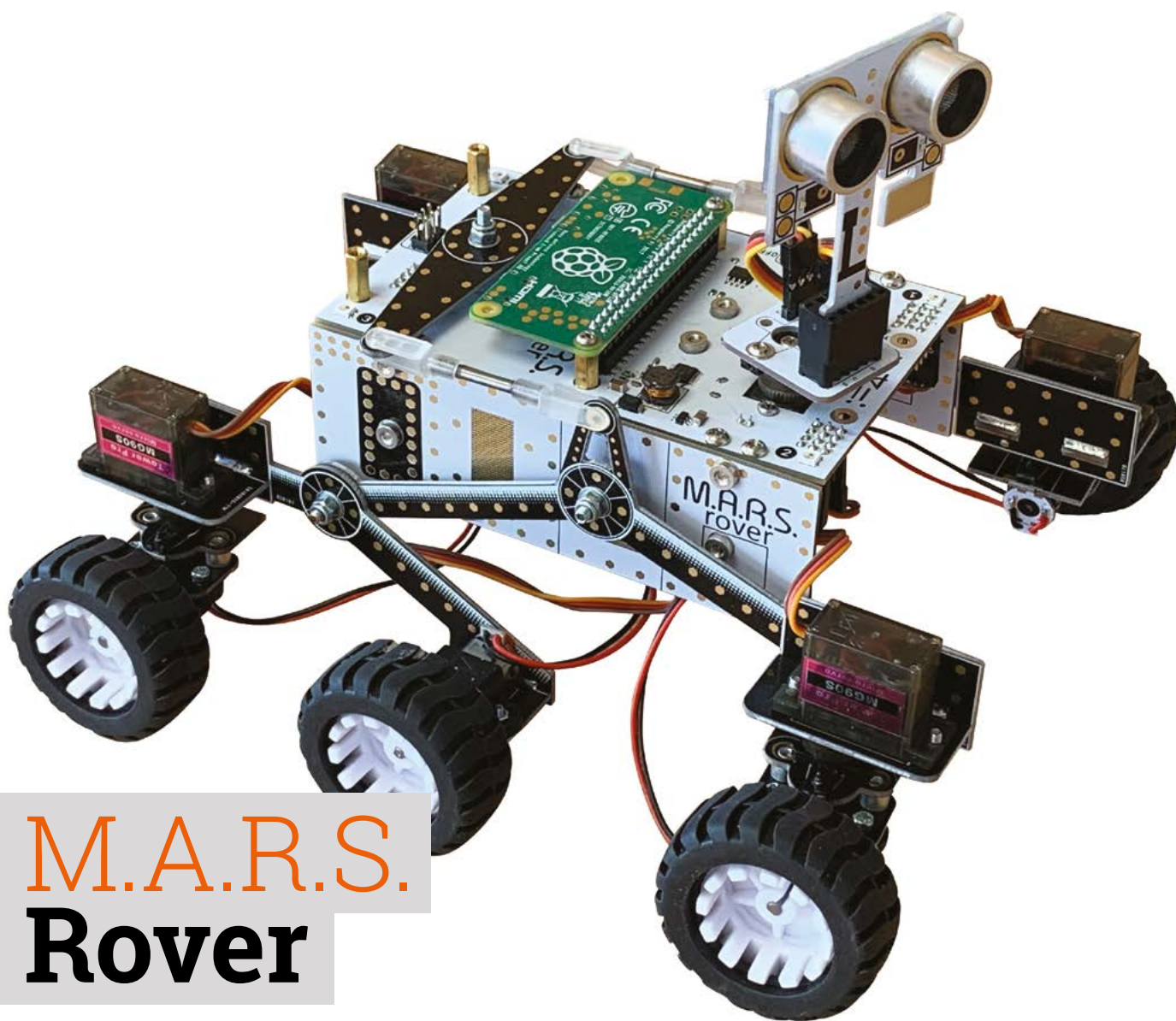
We would have loved this to bits as kids. And let's face it, that's what Raspberry Pi is all about. 

▲ It may not be the thinnest laptop, but CrowPi2 is packed with goodness

Verdict

We're taken with CrowPi2. Get past the rough exterior and you'll discover an absolute trove of components and educational material. Wonderful stuff.

9/10



M.A.R.S. Rover

► 4tronix ► magpi.cc/marsrover ► £120 / \$157

Explore new territory with this sophisticated six-wheeled robot kit. By **Phil King**

▲ This six-wheeled Rover even features a rotating mast with ultrasonic distance sensor

As we don't have our own spaceship, we aren't be able to test out 4tronix's M.A.R.S. Rover on another world, but it's an impressive vehicle featuring six driven wheels, four of them steerable, and a unique suspension system.

With no fewer than three space missions heading for the 'red planet' this summer, Mars is a hot destination right now. Traversing its rocky landscape will be two new rovers, including NASA's Perseverance Rover.

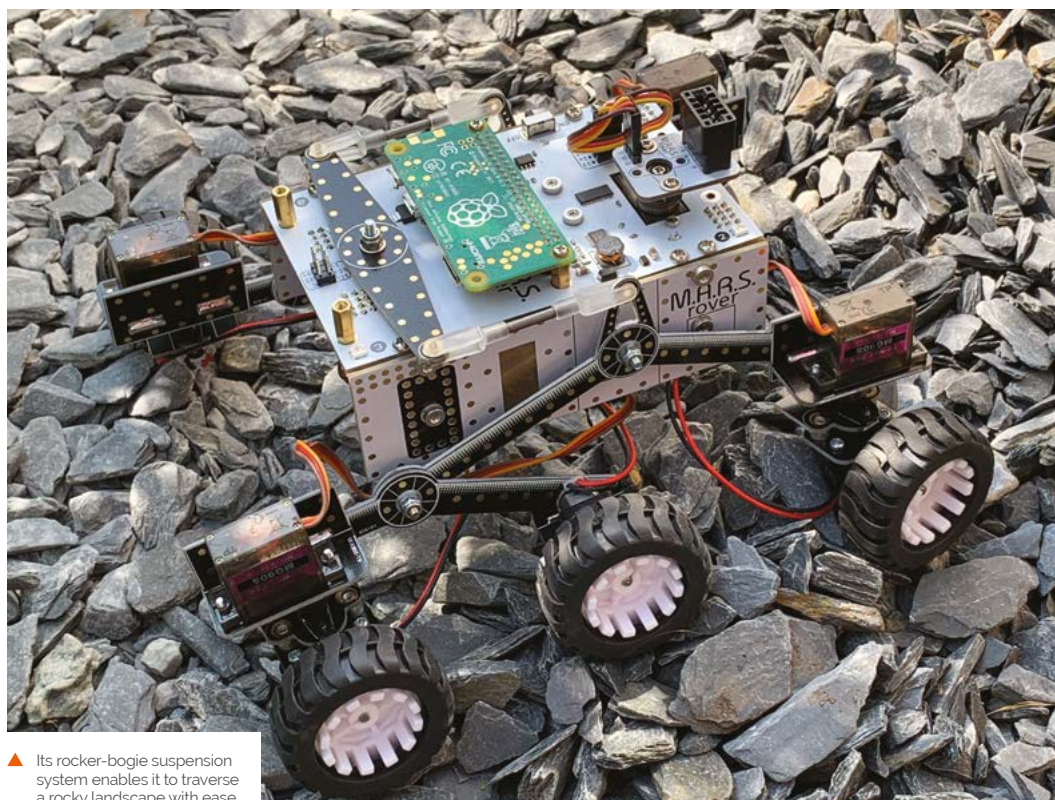
Like its predecessors, including Curiosity, Perseverance features a rocker-bogie suspension system that enables it to clamber over large rocks without tipping over. A similar system is used by

the 4tronix M.A.R.S. Rover, enabling it to tackle tough terrain. To help avoid larger obstacles and walls, the Rover features a rotating mast with ultrasonic sensor.

Start building

Packed in a cardboard box, the kit features a large array of components, including six motors, five servos, and no fewer than 30 PCB shapes to create the structure. In large kits like these, finding the right components and tiny screws can be a pain, so it's a great help that they're sorted into ten plastic bags relating to each step of the assembly process.

The latter is detailed in an online guide featuring plenty of photos, making it easy to follow. Even



▲ Its rocker-bogie suspension system enables it to traverse a rocky landscape with ease

so, you'll need plenty of time to build such a serious kit. 4tronix reckons it will take even an experienced kit builder around two hours; for others, you can at least double that time.

In the event, it took us several hours to assemble, partly due to an issue with screwing the bogies onto the rocker arms on either side using a small screw and nyloc nut. We found it impossible to screw it on tightly enough using the supplied mini screwdriver and spanner, so had to resort to using a larger, sturdier screwdriver.

Other than that hiccup, it was a fairly straightforward if lengthy process. Some of the PCB pieces have nuts pre-installed, which does make things a lot easier.

Steering servos

Four servos are used to steer the front and rear wheels. A handy little ServoZero board is included to connect to the servos during assembly and set them to zero before attaching the horn and motor mount at approximately 90 degrees to the Rover's body. Fine adjustments may be made to the wheel alignment later in the software.

A differential system, connecting the rocker arms, keeps the rover body level as the wheels on one side or the other traverse uneven terrain.

Fitting the motors is easy enough, as each is pre-wired with a JST connector that fits into one of the ports on the main board.

With the main body and suspension system assembled, and chunky wheels attached, the final touch is to add a mast with an ultrasonic distance sensor to the top. This even has a servo so you can rotate it with code.

“Featuring six driven wheels, four of them steerable, and a unique suspension system”

Rolling Rover

Insert four AA batteries (not supplied) in the holder at the bottom the main board and mount a Raspberry Pi Zero on top and you're ready to install the software. A Python library is accompanied by several code examples, including one that enables you to control the Rover remotely (via SSH) with your keyboard. There's also an example to test an optional keypad (£9/\$12) that fits onto the rear.

While the Rover moves at a fairly sedate pace, the steering servos allow it to turn quickly and more smoothly than with the typical tank steering on other wheeled robots. Best of all, that rocker-bogie suspension system enables it to roll over an obstacle, even when only one set of wheels is traversing it, with all six wheels magically maintaining contact with the ground – it's fascinating to watch in action and far superior to fixed-wheel buggies at handling difficult terrain. **M**

SPECS

MOTORS:

6 × 80 rpm 6V, N20 micro gear motors

SERVOs:

5 × MG90S metal gear analogue micro servos

POWER:

4 × AA battery pack (batteries not supplied)

FEATURES:

Ultrasonic distance sensor, RGB LEDs, white LEDs, rocker-bogie suspension

DIMENSIONS:

200 mm (L) × 185 mm (W) × 170 mm (H, with mast)

Verdict

A little pricey, but this is a high-quality kit. The Rover looks cool and can traverse the toughest terrain with its six driven wheels, servo steering, and clever suspension system.

9/10

10 Best:

Starter kits

Get started with your Raspberry Pi journey

Whether you've got a Raspberry Pi or not, it can sometimes be tricky to figure out where to start. There's a whole host of starter kits for people new to Raspberry Pi – here are ten of our favourites. [M](#)



▲ Official Raspberry Pi Desktop Kit

Everything you need

The official kit comes with a Raspberry Pi, an official keyboard and mouse, an official power supply, an official case, and *The Official Raspberry Pi Beginner's Guide* book, all in one neat little package.

£117 / \$120 | magpi.cc/desktopkit



▲ Pirate Radio

Internet and sound

This cool kit from Pimoroni is not only a neat little portable internet radio, it's also a great way to learn about basic internet usage and connecting sound to a Raspberry Pi Zero.

£43 / \$47 | magpi.cc/pirateradio

► CamJam EduKit 3

Robots for beginners

We talk about this kit a lot, but for good reason – it's a fantastic (and cheap) jumping off point for learning about Raspberry Pi robotics. There are also some nice 3D-printable chassis for it.

£18 / \$23 | magpi.cc/edukit3



◀ AIY Vision Kit

Updated sequel

This kit is a partner to the Voice Kit, and does image recognition using AI. We've seen some fun stuff made with image recognition, and this kit is very hackable once you've learnt all about it.

£83 / \$110 | magpi.cc/aiyvision



▲ CamJam EduKit

Electronics for beginners

This little kit will help you get started with electronics and digital making. It includes a breadboard, basic components, and jumper wires, and comes in a handy pocket-size tin for taking to Raspberry Jams.

£5 / \$7 | magpi.cc/camjamkit

► AIY Voice Kit v2

Updated classic

You may remember this from the cover of issue 57 of *The MagPi* – this updated version comes with its own Raspberry Pi Zero WH, and can help you get started with cool voice control.

£46 / \$60 | magpi.cc/aiyvoice



◀ Retro Gaming Kit

Get gaming

Everything you need to start doing some retro gaming with Raspberry Pi – including a Raspberry Pi itself! Check over our previous issues (and this one) for tips on how to play retro games with Raspberry Pi.

£88 / \$115 | magpi.cc/gamingkit



▲ Picade Console

Arcade-style gaming

A smaller version of the full Picade, this is more like a plug-and-play gaming device. You just need to add a TV and you'll be straight back to the arcades of the late eighties.

£75 / \$82 | magpi.cc/picadeconsole



◀ OctoCam

Sticky camera

This cute kit includes a Raspberry Pi Zero W, 5MP camera, and acrylic mount/stand. Have it guard your room or look out your window, thanks to its trademark cephalopodic suction arms.

£43 / \$47
magpi.cc/octocam

► Raspberry Pi 4 Starter Kit

Simpler starter kit

This smaller, simpler, and cheaper starter kit from The Pi Hut includes some basic accessories such as an HDMI cable, power supply, case, and a microSD card so you can get started with what you have.

£59 / \$77
magpi.cc/pihutstarter



Learn Scratch 3 with Raspberry Pi

This visual, block-based language is an ideal starting point for learning how to code. By **Phil King**

Raspberry Pi Projects

CREATOR

Raspberry Pi Foundation

Price:
Free

magpi.cc/scratchprojects

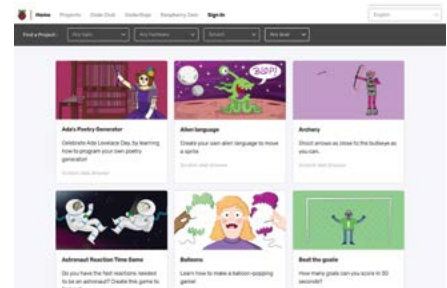
Created by the MIT Media Lab, Scratch has long been seen as an accessible entry point into the world of code, for children and adult newcomers alike. An excellent educational tool, it can be used to teach the core concepts of programming using an easy-to-use visual interface featuring drag-and-drop code blocks that are assembled to build programs.

Code scripts can be added to multiple sprites that appear on the 'stage', which makes it easy to start creating games and interactive stories. The latest version, Scratch 3, features a

slick new interface, along with extra blocks, sprites, and extensions.

While you can use the web interface at scratch.mit.edu, a desktop version of Scratch 3 comes pre-installed in Raspberry Pi OS that also features additional extensions to control a Sense HAT and electronic components connected to the GPIO pins.

The Raspberry Pi Foundation's learning resources website features a wide range of Scratch



projects to get you started, including some physical computing ones. The array of projects can be filtered by topic, hardware, and level. So it's easy to find exactly what you're after.

Books

Start reading to learn to code with Scratch

CODE CLUB BOOK OF SCRATCH

While written for Scratch 2, Code Club's book is a great primer. Big and colourful, it has easy-to-follow projects (requiring only minor changes to work in Scratch 3) and comes with stickers and magic glasses.

magpi.cc/ccbook1

25 SCRATCH 3 GAMES FOR KIDS

Written by teacher Max Wainwright, this colourful book is aimed at younger children.

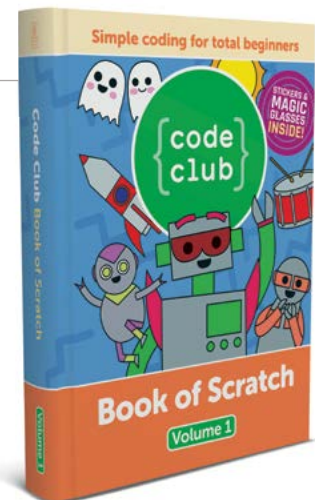
The projects are fairly short, but readers are encouraged to create similar games using the skills they've learned.

magpi.cc/25scratch

CODING FOR KIDS IN SCRATCH 3

With colourful illustrations and a dash of humour, this book aims to teach kids coding concepts in a fun way. Ten Scratch projects take you through using loops, variables, conditionals and more.

magpi.cc/kidsscratch3



The Official Scratch Coding Cards

CREATOR

Natalie Rusk

Price:
£15 / \$25nostarch.com/scratchcards3

If you're looking to teach youngsters to learn Scratch, this set of 76 flash cards – now updated for Scratch 3 – is a neat learning aid, breaking simple projects down into easy-to-follow steps. While a little pricey, it does feature sturdy, colourful cards.

The front of each card depicts an activity – such as Pong, Write an Interactive Story, Create a Virtual Pet, and Play Hide and

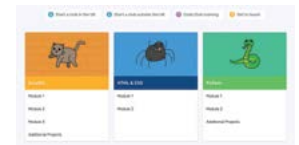


Seek – while the back shows how to put code blocks together to make that project come to life.

Along the way, kids learn core coding concepts like sequencing, conditionals, and variables. While some projects require the user to follow the cards in a set sequence, others give you the freedom to experiment by using the cards in any order.

Online resources

Check out these sites and tutorials



CODE CLUB

While many of the projects also appear on the main Raspberry Pi Foundation site, there are extra ones here and they're arranged in modules for different skill levels. codeclub.org

SCRATCHED

While this official community for Scratch educators is now closed to new contributions, the archive is packed with useful information and resources for teaching – and learning – the popular block-based language. magpi.cc/scratched

SCRATCH TUTORIALS

Don't forget that Scratch itself offers a built-in series of tutorials to help you get started and master a wide range of techniques, from spinning sprites to handling keyboard controls. Just click on the Tutorials button in the top menu bar of a project. magpi.cc/scratchtutorial

Coding Games in Scratch

CREATOR

Jon Woodcock

Price:
£15 / \$20magpi.cc/dkscratch

Now updated for Scratch 3, this has long been one of the most popular books for teaching children how to code in Scratch. It is also available in rebranded form in the UK, as part of the 'Computer Coding for Kids' Carol Vorderman series.

Following the step-by-step guides, readers get to learn Scratch by building games, ranging from basic ones such as Star Hunter (drag a sprite around) and Cheese Chase (maze) to more complex affairs like the two-player Glacier Race.

The instructions are clear and easy to follow, accompanied by plenty of screen grabs and diagrams to help explain key

concepts such as collision detection. Each chapter also ends with a 'hacks and tweaks' section giving the reader ideas on how to customise their creation. magpi.cc/dkscratch





Alan O'Donohoe

One of the original Raspberry Jams celebrates its 100th meeting. We talk to one of its organisers

➤ Occupation **Specialist Leader in Education** | ➤ Community role **Raspberry Jam organiser**
➤ Twitter **@PrestonRJam**

Getting to 100 of anything is always cause for a celebration, although we may be a bit biased about that at the moment. In the case of the Preston Raspberry Jam, which just celebrated its 100th event, it's even more important as it's one of the longest-running Raspberry Jams in the world, if not *the* longest.

"When the Raspberry Pi computer was first released, there was one group of people who wanted to know what they could do with it and another group that had lots of ideas, and another group that weren't really sure why people were getting

excited," says Alan O'Donohoe, one of the founding members of Preston Raspberry Jam. "We intended the Raspberry Jam to be a regular monthly gathering or club for like-minded people to come together and share projects, knowledge, and experience. It was actually Deborah, my wife, who coined the name 'Raspberry Jam' for the monthly gatherings."

Is there a particular flavour to activities at a Preston Jam?

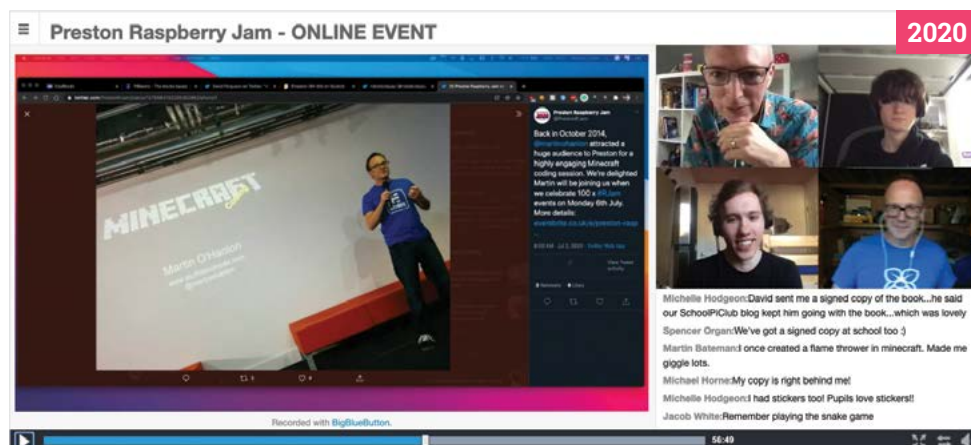
The activities include workshops, hands-on sessions, support, talks, project demonstrations, discussions about new hardware

and software, and something that one of our regulars Simon calls a 'gather round' where people gather round a table to learn more about something. In the early days, most projects had a very strong link to the Raspberry Pi computer, but less so nowadays. In recent years there has been more emphasis on low-cost and affordable technologies like micro:bit, Arduino, and Linux. We regularly have children lead workshops, or present lightning talks on a project they've been working on.

What are some of the challenges running an event like this for so long?

When we first started, we'd never expected that it would run every month for eight years or more. I think we were all amazed, to be honest, how many people would turn up month after month after month. We've had to move and adapt with the times and to suit younger audiences and changing interests. In the early days, people used to bring along a Raspberry Pi computer with them, asking for help to set it up, or to show us something they had done with it.

▼ The Centenary Raspberry Jam had a huge number of guests from around the Raspberry Pi community



2013



◀ The Preston Raspberry Jam attracted a lot of different people within its first year

▼ A young Josh Lowe presents at the Raspberry Jam – he'd go on to make EduBlocks

100 Jams of memories

"It has been incredible seeing so many children grow up with our Jam and take more of a lead in running activities. When Josh [Lowe] first started coming, he was just seven years old; now he regularly outshines the adults with his projects and achievements and he's run several Jams without Martin or myself. Cerys, who used to travel to our Jams from Stafford every month (a three- to four-hour return trip), now hosts a very popular monthly Jam at Stafford Library and was awarded EU Young Digital Girl of the year award. Jay was four when he gave his first lightning talk at our Jams; now aged 9, he regularly runs robotics and programming workshops at our Jams and other tech community events.

"We recently introduced monthly themes, e.g. Computer Games, Robots, Wearables, Try This At Home. One of the highlights at our 'Data Driven' August Jam was Genevieve Smith-Nunes's presentation highlighting projects that linked dance movements and digital sensing technologies with algorithms and performance art."

How have attendees changed over the last eight years?

If you look at the photos of our first Jams, you'll see it was mostly adults with only a handful of children. After some parents suggested we start the Jam events earlier in the evening, we noticed more families attending. We tried some Saturday morning Jams in 2012, but we found these difficult to sustain as we had lots of children and very few adults able to support the activities. We have some people who only attend once or twice a year, and others who attend every month without fail.

What was the 100th Jam like in these odd times?

About a year ago we started talking about what we should do for our 100th Jam celebration, but our plans had to change when social distancing restrictions were introduced. Instead we invited [a group of] people who have been on our #RJam journey right from the

2014



start to join us for a series of panel discussions in an online jam. That was Martin [Bateman, Senior Lecturer, Computing, UCLan]'s idea and it worked an absolute treat. Since March we've been holding all our #RJam events online. We increased the frequency to two Jams a month, one with lightning talks and another with workshops. It's been incredible to see how our numbers have grown rapidly to about 100 every month as enthusiasts have joined us from areas way beyond Preston. We've had many attendees join us from Scotland that have long wanted to attend our Jams, but can now take an active part online. **W**


This Month in Raspberry Pi

MagPi Monday

Amazing projects direct from our Twitter!

Every Monday we ask the question: have you made something with a Raspberry Pi over the weekend? Every Monday, our followers send us amazing photos and videos of the things they've made. Here is a small fraction of them. Follow along at the hashtag #MagPiMonday. [M](#)

01. We're not sure whether to be sad or amazed by this cursed robot
02. The CyberPiDeck will soon be jamming with the console cowboys in cyberspace
03. We can't wait for this to be back up and running!
04. This little device uses the shipping forecast to generate music. Which, in our books, makes it extremely cool. Check it out here: magpi.cc/gxmsyh
05. We've loved seeing MABEL being built up over the last few... weeks? Months? Time doesn't have much meaning any more. Check out the blog: magpi.cc/qyypn
06. This time-lapse is quite pretty, and you can check it out here: magpi.cc/ppyiwr
07. A tiny little sentient robot would be a wonderful friend, don't you think?
08. Frogs, bubbles, and the internet. The perfect combination
09. We sometimes see these fun 3D scanner projects and think... we could actually use that



8 Bits and a Byte
@8BitsandaByte


01

Replying to @TheMagPi

We built a robot that toys with your emotions!

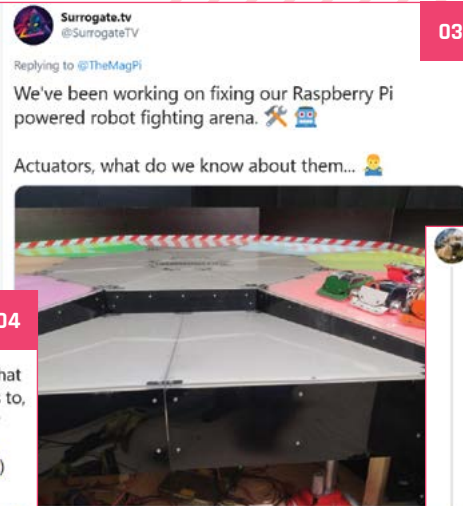
The project detects your emotional state using the AIY Vision Kit. To get your mood back to neutral we use a speaker to tell you facts, either fun or depressing ones. Like 'Cows have best friends', or 'everything you love will die'

Mars is entirely populated by robots.





02



03



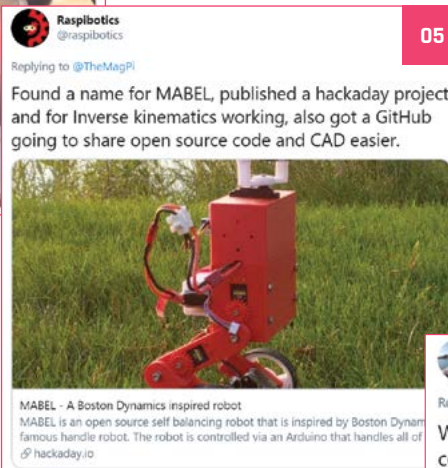
04



06



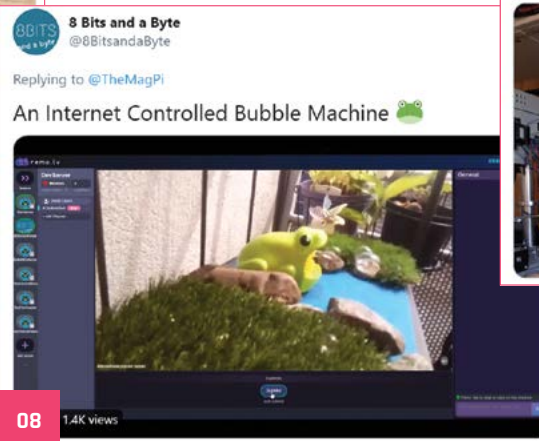
07



05



09



08

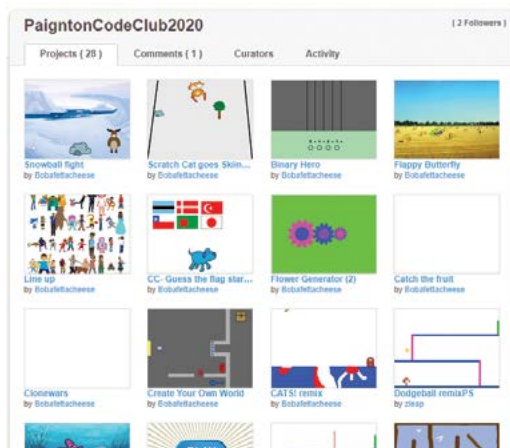
Running a virtual Code Club

Tips from the Paignton Library Code Club on how to go online

**JOIN
PAIGNTON
LIBRARY CODE CLUB!**

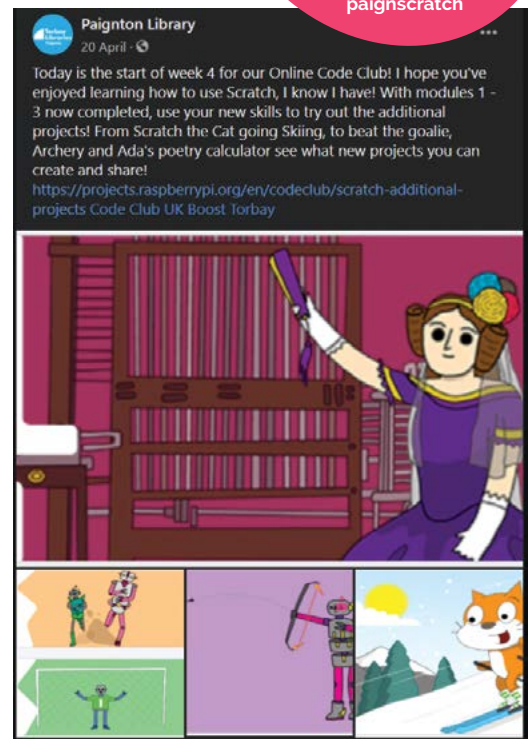
The Code Club is active 24/7, with new activities being sent out over Twitter from Paul (@zleap2018) and Paignton library (@paigntonlibrary). Check out their Scratch studio here: magpi.cc/paignscratch

During this year, a lot of in-person events have sadly been cancelled or postponed. For regular Code Clubs, CoderDojos, and Raspberry Jams, this has resulted in a new wave of online events. Paul Sutton is just one event organiser who has done this, with help from the original hosts at Paignton Library. Here's how they did it. [M](#)



Send out activities

"Between myself and [Charlotte Summer from Paignton Library], we have been sending out activities via Twitter (@zleap2018, @paigntonlibrary). We have also set up a Scratch studio so that anyone taking part has somewhere to share their work, at least for Scratch (magpi.cc/paignscratch). The idea was that anyone using the other activities could share back via social media links."



Make it personal

"I post items to my blog (magpi.cc/paulblog) before it is tweeted out. [It] adds a more personal touch than just sending links.

"I am also keen to promote other learning, e.g. courses via freeCodeCamp (freecodecamp.org), which can cover a wide range of topics, such as Python Networking tutorials. Again, I tweet the links to the library."



Crowdfund **this!** Raspberry Pi projects you can crowdfund this month



SnowPi

That time of year is rolling around, and Ryan Walmsley has you covered with a relaunch of his classic SnowPi kit – now with upgraded RGB lights for a more magical Christmas show. It's already been funded at the time of writing, so jump on if you want one!

► kck.st/3hJ9MTF



RaspyKeys

Unlike some teaching apps you'll find on smart devices and tablets, the point of RaspyKeys is to plug directly into a digital keyboard so it can read the data as you play. It can also then control the keyboard. It's a very interesting idea.

► kck.st/2BGAXsf

CROWDFUNDING A PROJECT?

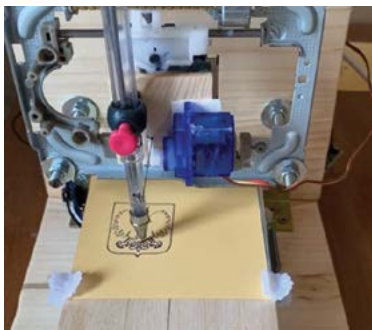
If you've launched a Raspberry Pi-related project, let us know!
magpi@raspberrypi.org

Best of the rest!

Here are some other great things we saw this month

DRAWING MACHINE

This upcycled little drawing robot uses an old CD drive to control some of the motor functions of the drawing!



► magpi.cc/pgamps

SNES CONVERSION

We haven't seen one of these projects in a little while, but it's great to see people are still installing Raspberry Pi boards into old consoles.



► magpi.cc/cwh5nt

UNBRICKING A BIKE

A weird modern problem is smart devices just ceasing to work properly because they're no longer supported. When this person's smart-bike-thing stopped working properly due to a legal battle, he fixed it with a Raspberry Pi Zero.



► magpi.cc/hxppzc

POSSIBLE EVENT CANCELLATIONS

Please follow local public health advice and take decisions on whether to cancel clubs or events in consultation with the venues that host them.

Raspberry Jam Event Calendar

Find out what community-organised Raspberry Pi-themed events are happening near you...

01. The Repair Café Experiment

📅 Saturday 5 September

📍 Online

▶ magpi.cc/5g7c2w

Find out how to repair, mend, make, hack, or repurpose your things at home online.

02. Medway Makers Tech Meetup

📅 Sunday 6 September

📍 Online

▶ magpi.cc/aClcwv

Workshops in building cool things with Arduino, how to code for beginners, how to use a Raspberry Pi, and so on.

03. Houston Raspberry Pi Meetup

📅 Wednesday 9 September

📍 Online

▶ magpi.cc/W2Ugxx

A community of coders, artists, educators, and engineers interested in project-based learning with Raspberry Pi.

04. Hackaday U: Linux + Electronics: A Raspberry Pi Course

📅 Friday 11 September

📍 Online

▶ magpi.cc/o0gDU9

This course is for those who had wanted to go from Arduino to a much more powerful Raspberry Pi.

05. GGTP 94: Crafter-Zoom!

📅 Sunday 20 September

📍 Online

▶ magpi.cc/E5wAsc

A 'Crafter-noon' (their words) for all kinds of makers and crafters, from knitting to soldering.

06. Raspberry Pi Teacher Training Workshop

📅 Wednesday 23 September

📍 Online

▶ magpi.cc/HZrUPL

Hands-on introduction to Raspberry Pi for primary and secondary teachers to support and enable participation in the Astro Pi Mission Zero.

07. OP209: HYPERSIM Advanced Applications and Optimizations

📅 Monday 28 September

📍 Online

▶ magpi.cc/K4nPSq

In the HYPERSIM Advanced Applications and Optimizations training, users will learn techniques for test automation, CPU, and FPGA simulation.

08. BCG Platinium Hackathon 2020

📅 Friday 23 October

📍 Online

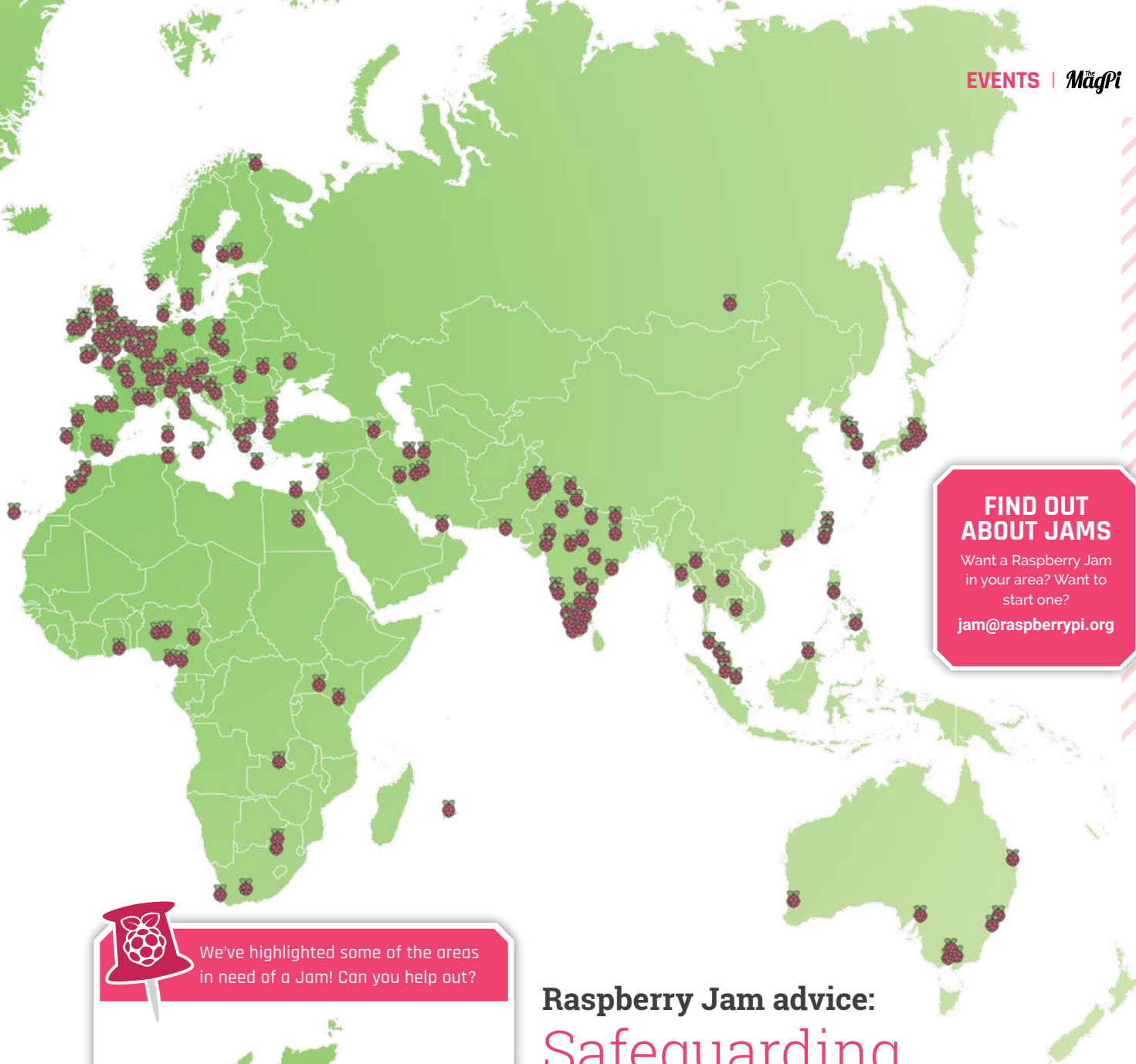
▶ magpi.cc/4BD3Br

Several cities, one goal: coding strategies that shape the future, unlock digital potential, and ignite change!

FULL CALENDAR

Get a full list of upcoming events for September and beyond here:

rpf.io/jam



FIND OUT ABOUT JAMS

Want a Raspberry Jam in your area? Want to start one?

jam@raspberrypi.org



We've highlighted some of the areas in need of a Jam! Can you help out?



Raspberry Jam advice: Safeguarding

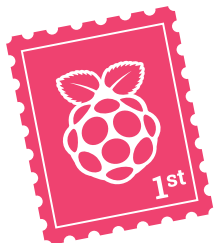
"Although most of our volunteers are DBS-cleared through other voluntary work, we made a conscious decision that we wanted parents to remain with their children. The responsible adult does not need their own ticket, but requiring parents or guardians to accompany their children removes the need for DBS paperwork, and means that every child has a familiar authority figure on hand. We also find that children are more focused in the workshops when their parents are around."

Andrew Oakley – Cotswold Jam

Every Raspberry Jam is entitled to apply for a Jam starter kit, which includes magazine issues, printed worksheets, stickers, flyers, and more. Get the book here: rpf.io/guidebook



Your Letters



RAW photos

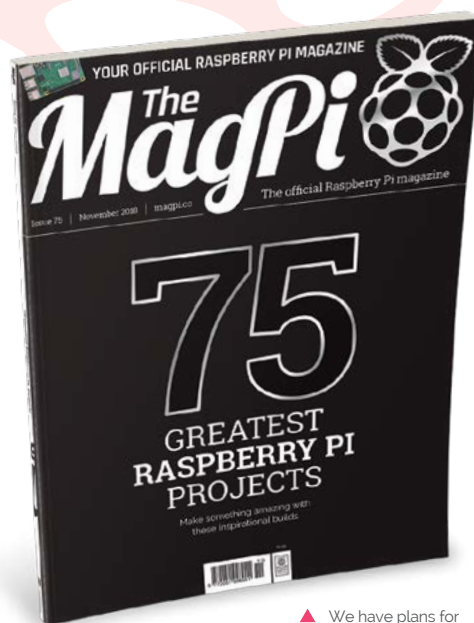
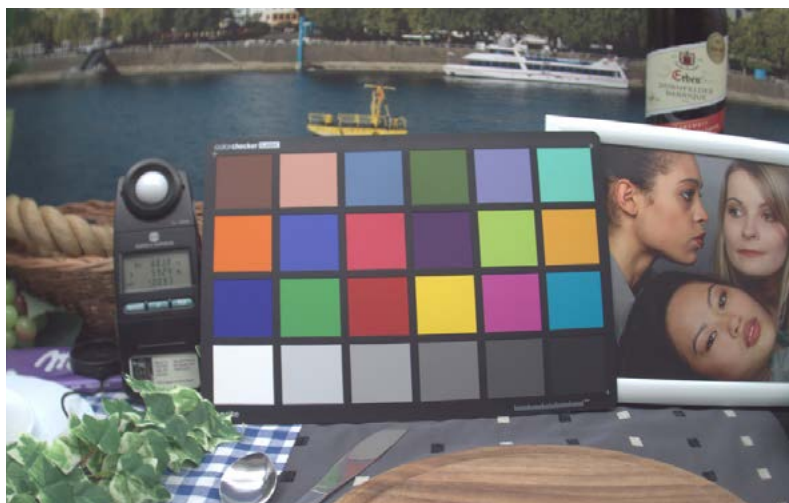
I got the High Quality Camera recently and so far it's been quite nice to use. My partner has a background in photography and is wondering if there's any way to get the RAW image from the camera rather than JPG? Apparently these can be a lot better for processing and getting nice images out of. Is there any thing you can point me towards?

Tea via Facebook

Outputting images as RAW files is not something you can do with the standard Raspberry Pi raspistill software, but there are methods to get it to output a DNG RAW file for post-processing! Check out this Raspberry Pi blog on how to get them: magpi.cc/hqraw.

RAW processing always seems like a dark magical art to us, although the results are undeniably excellent when you know what you're doing.

▼ Colour charts and white balance sheets are great for colour correction and post-processing



▲ We have plans for issue 100, and we'll let you know them shortly!

Article database

It would be really nice to have an index for the tutorials. Scrolling through 96 pages gets tough, and you have to make notes of pages if you want to re-review similar projects.

Henry via email

This is something we've been wanting to do for a while now – there used to be some fan ones around but they're currently a bit defunct. As we're coming up to our 100th issue, we've been thinking that we should finally set up an official index for the community to make searching for articles easier. Once we've got it figured out and implemented, we'll mention it in the magazine and on our social feeds so you can use it and help us fill it out. Stay tuned!



◀ Check out *Get Started with Arduino* from our colleagues at HackSpace magazine!

Best tool for the job

Since the original Raspberry Pi came out (and I am still using these), I have played with many ideas, put together several devices, and had a great time. My latest Raspberry Pi is a Raspberry Pi 3 – don't think I need a Raspberry Pi 4 yet, but you never know.

The way Raspberry Pi moved me initially was back into my love and enjoyment of electronics together with computing, which the disturbance of making a living had so rudely interrupted.

For some time Raspberry Pi was the way, the pathfinder.

Whenever I finished, and left running, a project, I worried about power and SD card possible problems and have introduced UPS (uninterruptible power supplies), which seem to have worked with the exception of my Free PBX, (Asterisk) which seems to have a regular SD meltdown. (SD backups, what a good idea?) Luckily, it is local so I can get to fix it simply.

In the last couple of years my projects have led me to the Arduino and NodeMCU, ESP8266-type devices.

For small, simple projects they have been great. Raspberry Pi still covers the complex stuff and all my original projects, bar one, still sit wonderfully on Raspberry Pi boards.

I now have temperature sensors littered around the home, door monitors for the house, and even remote web servers for temperature sensing in a remote home.

I could have used Raspberry Pi for these, but I think the correct choice for these was not to do so.

I do, of course, use one of my Raspberry Pi systems running Mosquitto as an MQTT sever to report all the data and set in motion actions.

Raspberry Pi took me on a wonderful journey and one that has, I think, fulfilled the original reason the computer was developed.

I look forward to many more projects, where does it end?

Each project I will try to choose the best device; some will be Raspberry Pi, but many others not.

Peter via email

You're absolutely right, that's one of the big things about Raspberry Pi – getting people interested and introduced to electronics, computing, and digital making. This kind of hobby can be a journey, building up your skills and finding what's right for a specific job as you go.

We've done a few features on using Arduino with Raspberry Pi as well, so it's definitely something we support.

Contact us!

- ▶ Twitter **@TheMagPi**
- ▶ Facebook **magpi.cc/facebook**
- ▶ Email **magpi@raspberrypi.com**
- ▶ Online **raspberrypi.org/forums**

THE BEST-SELLING MAG FOR PC HARDWARE, OVERCLOCKING, GAMING & MODDING

PC HARDWARE ENTHUSIASTS



ISSUE 205 OUT NOW

VISIT **CUSTOMPC.CO.UK** TO LEARN MORE

WIN ONE OF FIVE PICAR-V V2.0 ROBOT KITS!

We reviewed the PiCar-V kit last issue (magpi.cc/96) and we thought it was pretty cool. Now is your chance to win one!

"PiCar-V is an open-source robot learning kit based on Raspberry Pi. Equipped with a wide-angle USB webcam, it is powerful together with three whole new circuit boards and less, but simpler, wiring."



Head here to enter: magpi.cc/win

Learn more: magpi.cc/picarv

Terms & Conditions

Competition opens on **26 August 2020** and closes on **23 September 2020**. Prize is offered to participants worldwide aged 13 or over, except employees of the Raspberry Pi Foundation, the prize supplier, their families, or friends. Winners will be notified by email no more than 30 days after the competition closes. By entering the competition, the winner consents to any publicity generated from the competition, in print and online. Participants agree to receive occasional newsletters from The MagPi magazine. We don't like spam: participants' details will remain strictly confidential and won't be shared with third parties. Prizes are non-negotiable and no cash alternative will be offered. Winners will be contacted by email to arrange delivery. Any winners who have not responded 60 days after the initial email is sent will have their prize revoked. This promotion is in no way sponsored, endorsed or administered by, or associated with, Instagram or Facebook.

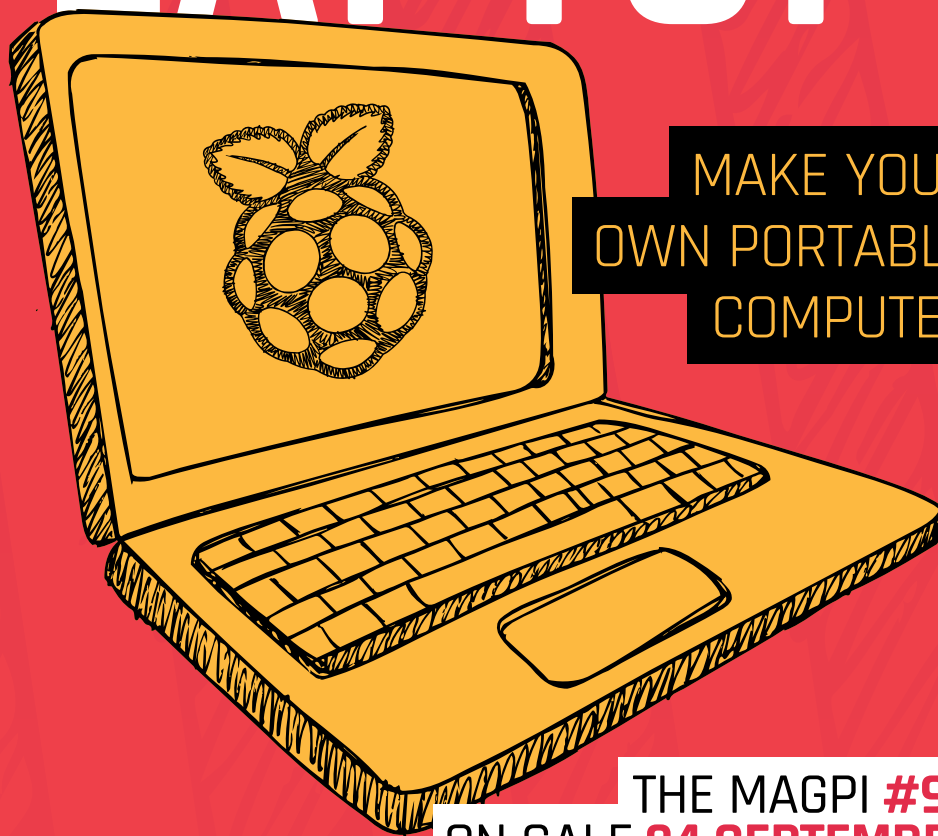
Wireframe

Join us as we lift the lid
on video games



Visit wfmag.cc to learn more

BUILD A RASPBERRY PI 4 LAPTOP



MAKE YOUR
OWN PORTABLE
COMPUTER

THE MAGPI #98
ON SALE 24 SEPTEMBER

Plus!

Run a Raspberry Pi
from the cloud

Get ready for
Halloween

Game with a retro TV
arcade machine

DON'T MISS OUT!

magpi.cc/subscribe

TWITTER @TheMagPi

FACEBOOK fb.com/MagPiMagazine

EMAIL magpi@raspberrypi.com

EDITORIAL

Editor

Lucy Hattersley
lucy@raspberrypi.com

Features Editor

Rob Zwetsloot
rob.zwetsloot@raspberrypi.com

Sub Editors

Phil King and Nicola King

ADVERTISING

Charlotte Milligan
charlotte.milligan@raspberrypi.com
+44 (0)7725 368887

DESIGN

criticalmedia.co.uk

Head of Design

Lee Allen

Designers

Sam Ribbits, Ty Logan

Illustrator

Sam Alder

CONTRIBUTORS

Wes Archer, Mike Cook, David Crookes, PJ Evans, Rosemary Hattersley, Nicola King, KG Orphanides, Tim Richardson, Mark Vanstone

PUBLISHING

Publishing Director

Russell Barnes
russell@raspberrypi.com

Director of Communications

Liz Upton

CEO

Eben Upton

DISTRIBUTION

Seymour Distribution Ltd
2 East Poultry Ave,
London EC1A 9PT
+44 (0)207 429 4000

SUBSCRIPTIONS

Unit 6 The Enterprise Centre
Kelvin Lane, Manor Royal,
Crawley, West Sussex, RH10 9PE
+44 (0)1293 312193
magpi.cc/subscribe
magpi@subscriptionhelpline.co.uk



This magazine is printed on paper sourced from sustainable forests and the printer operates an environmental management system which has been assessed as conforming to ISO 14001.

The MagPi magazine is published by Raspberry Pi (Trading) Ltd, Maurice Wilkes Building, St John's Innovation Park, Cowley Road, Cambridge, CB4 0DS. The publisher, editor, and contributors accept no responsibility in respect of any omissions or errors relating to goods, products, or services referred to or advertised in the magazine. Except where otherwise noted, content in this magazine is licensed under

a Creative Commons Attribution-NonCommercial-ShareAlike 3.0 Unported (CC BY-NC-SA 3.0).
ISSN: 2051-9982.





Building your skills

Practice makes perfect, but working your way up to a big project is just as important, according to **Rob Zwetsloot**

In my circles of Twitter, I see about once or twice a year a joke that goes something like ‘What do you mean I have to work at something to be better at it?!’

It’s part of the ‘relatable content’ subgenre of jokes that my generation like to tell, in this case a vocalised realisation presented as fake outrage. I do find it relatable as well – my attention can easily wane if I’m not immediately good at something, even if it’s something I enjoy doing.

Now, generally there are socio-political factors involved with some of this which boil down to your brain thinking ‘if you can’t do it well, it’s not worth doing because capitalism’, which far smarter people than me have written about at length. In the case of digital making and coding and building, it’s definitely worth understanding that beyond practice, there’s learning multiple different skills. And that feels a little more agreeable to the ol’ brain gremlins.

One project at a time

As well as the tutorials I sometimes write for *The MagPi*, I also have buildery/codery hobbies such as making robot model kits and creating games. I currently have big, lofty

aspirations for what I can accomplish in both of these areas; however, I’ve fast come to the realisation that I’m not going to be achieving them quickly. This custom Gundam robot I’d like to make requires painting, weathering, customisation, kit-bashing, and more. Each of which is its own skill that I need to work on.

“ Treat yourself and leave some time free to make something just for fun ”

In this case, I’m able to build other kits – something I enjoy anyway – and maybe practise the painting aspect this time or have a go at weathering on the next kit.

It’s the same with digital making, with coding, wiring, woodwork, 3D printing, and more that you may need to work on separately before combining it all together to do, well, whatever it is you want. Robot obstacle course? A fully automated house? Fire-breathing dragon van? It’s all possible.

Doing things for fun

Beyond that, it doesn’t matter if you have a big goal to work towards.

In times like these, I think it’s very important to remember that

sometimes you can just make or do stuff because you’d like to. Creative outlets and hobbies come in all shapes and sizes, including digital making, and can be so rewarding. I created a little scrolling countdown clock using an 8×8 LED screen once, just because I wanted to see if I could. It was quite simple (thank you, GPIO

Zero) and a bit of fun to make at the same time. I ended up modifying it and using it as a proper countdown clock, but that doesn’t always have to be the outcome.

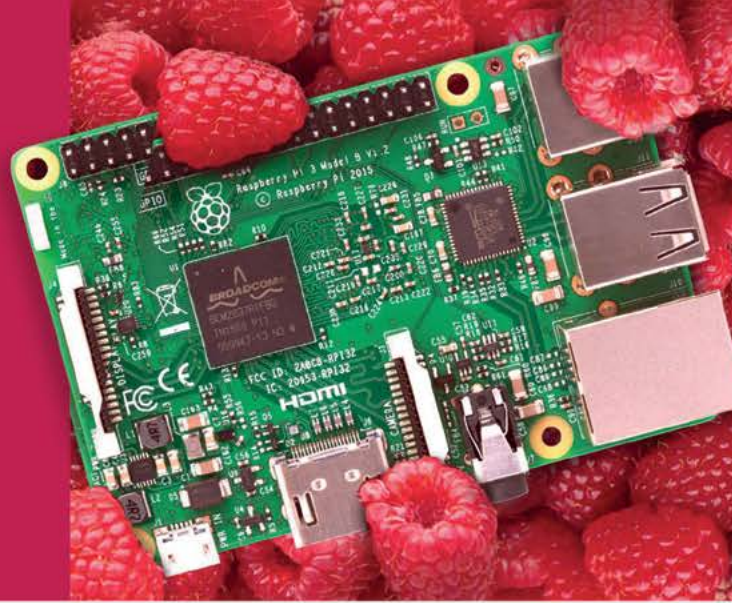
So treat yourself and leave some time free to make something just for fun. Upload it to the internet if you want. As long as you’re feeling fulfilled by the end of it, that should be all you need. **W**

AUTHOR Rob Zwetsloot

Rob is amazing. He’s also the Features Editor of *The MagPi*, a hobbyist maker, cosplayer, comic book writer, and extremely modest.

magpi.cc

American Raspberry Pi Shop



- Displays
- Cases
- Project Kits
- Add-on Boards
- HATs
- Arcade
- Cameras
- Cables and Connectors
- Sensors
- Swag
- Power Options
- GPIO and Prototyping

Partner and official reseller for top Pi brands:



and many
others!

Price, service, design,
and logistics support for
VOLUME PROJECTS





CHECK OUT THE NEW OKDO PROJECT HUB!

Find out more at www.okdo.com/projecthub



Find inspiring projects created by our customers and engineers that show you how to build and programme cool projects and how to get started with some of your favourite boards!

Find out more about our
full product range at:

www.okdo.com

